

**CONPAS:
Application of Knowledge-Based System
to Construction Planning**

by
Ariyoshi Miyaji and John Kunz

**TECHNICAL REPORT
Number 91**

January, 1994

Stanford University

Copyright © 1994 by
Center for Integrated Facility Engineering

If you would like to contact the authors please write to:

*c/o CIFE, Civil Engineering,
Stanford University,
Terman Engineering Center
Mail Code: 4020
Stanford, CA 94305-4020*

SUMMARY
CIFE TECHNICAL REPORT #91

Title: **CONPAS: Application of Knowledge-Based System to Construction Planning**

Authors: Ariyoshi Miyaji, John C. Kunz

Date: Feb., 1994

Funding: Obayashi Corporation

1. **Abstract:** This report describes a knowledge-based computer tool for construction planning, one of the crucial stages for a construction project. The objective of this research is to demonstrate a successful application of a computer tool named CONPAS (CONstruction Planning ASSistant). CONPAS is designed to produce erection plans for structural steel. The test cases show that CONPAS can provide erection plans including resources, task sequence and lay out of resources that the projects require for their completion. CONPAS shows that a knowledge-based tool can help engineers build construction plans.
2. **Subject:** To build a construction plan, engineers must understand how the construction is executed. This research attempts to analyze and describe a construction plan with five factors: components, actions, resources, task sequence, and lay out. CONPAS builds a plan using experiential knowledge that human engineers often employ for construction planning. Also, it links with CAD to obtain site constraints, show site lay out, and show plan execution by simulating the erection process.
3. **Objectives/Benefits:** This research addresses three problems that the construction planning phase currently has. First, CONPAS can be a good manual for construction planning where they now have only a little knowledge written in manuals. Second, this is a prototype computer tool to assist engineers in elaborating construction plans. Third, CONPAS helps the drawing-centered integration with construction planning.
4. **Methodology:** To represent a construction project, CONAPS adds the idea of task sequence and lay out to the three representations discussed in OARPLAN [Darwiche]. CONPAS is built on a commercial knowledge-based tool with a function to link CAD by UNIX interprocess communication. It uses experiential knowledge implemented in rules.
5. **Results:** This research plans the structural steel erection of the Terman Annex in Stanford. CONPAS provided steel erection plans including resources, task sequence, and lay out, that are considered feasible enough to be executed. It shows that the representations and the knowledge implementation might be appropriate. CONPAS also showed that the interactive operations with CAD were effective. This study exemplifies the effectiveness of application of a knowledge-based tool to construction planning. This research presents a software tool, CONPAS, and a video that explains how CONPAS builds a steel erection plan for the Terman Annex case.
6. **Research Status:** This research project, made as a study of a visiting fellow of CIFE, is complete. The further effort would be made in Obayashi to improve this prototype computer tool for practical use. Collection and implementation of more experiential knowledge is necessary. Also, an attempt to examine other kinds of work than steel erection should be made.

CONPAS: Application of Knowledge-Based System to Construction Planning

Ariyoshi Miyaji¹, John C. Kunz²

¹ Obayashi Corporation Tokyo, Japan. C/o The Center for Integrated Facility Engineering, Stanford University, CA 94305-4020

² Senior Research Associate, The Center for Integrated Facility Engineering, Stanford University, CA 94305-4020

Table of Contents

Abstract.....	1
1. Introduction.....	2
1.1 Background.....	2
1.1.1 Construction planning.....	2
1.1.2 Computer integration.....	2
1.2 Problem Statement.....	3
2. Research Goal.....	4
3. Construction Planning Assistant.....	5
3.1 Purpose.....	5
3.2 Representation.....	5
3.2.1 Essential three representations.....	5
3.2.2 Task-Sequence.....	6
3.2.3 Lay-Out.....	6
3.2.4 Zone.....	6
3.2.5 Project.....	7
3.3 Reasoning.....	7
3.3.1 Experiential knowledge.....	7
3.3.2 Lay-out examination using grids.....	10
3.4 Software Tools.....	11
3.4.1 User interface.....	11
3.4.2 System interface.....	11
3.5 Test case.....	12
3.5.1 Demo example.....	12
3.5.2 Terman Annex building.....	13
4. Conclusions and Implication.....	14
4.1 Contribution of CONPAS.....	14
4.2 Implication for the future.....	14
References.....	16
Figure 1.1-1.....	17
Figure 3.1-1.....	18
Figure 3.2-1.....	19
Figure 3.3-1.....	19
Figure 3.2-2.....	20
Figure 3.4-1.....	21
Figure 3.5-1.....	22
Figure 3.5-2.....	22
Figure 3.5-3.....	23
Figure 3.5-4.....	24
Figure 3.5-5.....	25
Table 3.3-1.....	26
Appendices.....	27

Abstract

This research discusses application of a knowledge-based computer tool to construction planning, one of the most crucial stages in a construction project. Scores of computer tools have been developed to assist civil engineers with construction planning. However, they are rarely focused on how to execute a construction project. They focus on the scheduling and the cost estimation of a project. The objective of the research is to demonstrate the possibility of successful application of knowledge-based computer technology to improve the productivity to plan execution of construction projects. CONPAS (CONstruction Planning ASsistant), includes: 1) an appropriate representation of a construction project in a computer tool, 2) implementation of the expertise of human engineers regarding construction planning, 3) realization of effective interfaces matching the characteristics of the planning work. A realistic representation of a project is indispensable for a computer tool dealing with construction planning. CONPAS attempts to represent a project with five constituents: Component, Action, Resources, Task-Sequence, and Lay-Out. CONPAS creates a plan inferring zones, resources, task-sequence, and lay-out. The experiential knowledge of engineers is successfully implemented in CONPAS to infer those four items. To realize an effective interface, CONPAS employs an interactive linkage of a symbolic model with 3D CAD. It enables users to view construction plans with 3D visualization. Test cases have resulted in feasible construction plans that could be executed. This prototype system shows that, at least for simple test cases, it is possible to integrate graphic models with symbolic engineering models, and that such integration helps engineers to build good construction plans.

1. Introduction

1.1 Background

1.1.1 Construction planning

This research addresses the process of construction planning, an essential phase of a construction project. A construction plan describes how to execute a construction and shows feasible alternatives for the execution.

The construction planning phase is important because civil engineers can comprehend objectives and methods more clearly by developing practical construction plans. As a construction plan is developed, engineers sometimes realize some tasks with specified equipment and materials that had not been expected to accomplish the project. When an engineer finishes a detailed construction plan, the scope of a project is clearly defined. The clear definition of the scope of a project must be done at the beginning of a project prior to the development of either the budget or the schedule [Oberlender]. The construction planning phase is important not only to comprehend the scope, but also to assume that execution will have sufficient quality and safety. A poor construction plan or an ill-balanced construction plan may cause accidents, environmental destruction, and an excess over the budget [Shokoku-sha]. A better construction plan leads the construction execution to a more satisfying result.

From the project management point of view, a construction plan includes the schedule and the cost estimation. However, it is crucial to have a plan of how to execute a construction before the schedule and the estimation are produced. As mentioned at the beginning, construction plan presents alternatives on how to execute a construction.

A "construction plan" includes the following items:

1. Objective of a project
2. Actions a project should include
3. Resources a project should employ
4. Sequence of activities a project should take
5. Spatial lay out of resources over time

Producing a construction plan, in this paper, means creating several feasible sets of those five items as well.

1.1.2 Computer integration

For several years, the construction industry has been trying to implement computer technology to improve productivity and profitability. In the area of construction, there is also a steady movement toward computer integration. This paper describes a technique for computer integration of a construction project work with 3D CAD drawings.

Figure 1.1-1 shows a common task flow of a construction project. It explains that the usual sequence of tasks of a construction project can be regarded as a circulatory flow, sometimes a series of circles or a spiral. A construction project begins with the marketing phase. After the proposal from a client, in the programming phase, the draft of the project is made along the client's concept. Through the design phase, a project gets concrete objectives for its completion by detailed drawings or specifications. Then, in the planning phase, the feasibility of the project is examined. A building or a bridge as a

product is completed in the execution phase. In the maintaining (operating) phase, a new proposal to improve the present condition is likely anticipated.

In a construction project, a drawing is one of the most effective vehicles to communicate with each participant of a project. A drawing contains a great amount of information in it and plays an important part in each of phases. From the marketing phase through the maintaining phase, drawings are in the center of each meeting, and they are changed or modified, then passed to the next stage of task. Thus, in a construction project, information goes around the task flow in the form of drawings.

In this paper, the computer integration is discussed as an effective and practical application of 3D CAD drawing throughout a construction project. Fruchter and Clayton discuss the importance of exploiting CAD software for the conceptual design stage of a project [Fruchter]. In the construction planning phase, the maximum CAD employment should be considered. The "drawing oriented" or the "drawing centered" integration is the basic idea for this research.

1.2 Problem Statement

To obtain more productivity for the tasks in the construction phase, there are three major problems producing construction plans.

First, for construction planning tasks, there are very few good manuals or textbooks available. Diversely scaled and specified projects make it difficult to establish a reliable theory to produce a construction plan. In addition, they often employ experiential or heuristic knowledge that does not fit into formulas.

Second, there are few computer tools to assist civil engineers with construction planning. Though there are computer tools for scheduling and estimation, they do not help to produce a construction plan from the beginning. The construction industry has been trying to implement computer technology, and the use of the technology is getting extensive. As they employ computer technology in other phases, e.g., computer aided designing in the design phase, or robotics in the execution phase, they should fully exploit computer technology in the planning phase as well.

Third, there is very little computer integration in the work process of construction projects. Throughout the work process of a construction project, from the marketing phase to the maintaining phase, however, there are specialized computer tools, but those tools are independent. Usually, they have to bring data from one tool to another, and some of the tools don't accept data from others. Hence, everybody must admit the necessity of integration either in a single phase or of multiple phases. Furthermore, computer tools being built in the future should adapt themselves to integrated systems.

2. Research Goal

This research is focused on building a computer tool with object oriented technology to resolve or ease those three problems mentioned in the previous chapter:

1. There is little knowledge of construction planning written in manuals.
2. There are few computer tools that assist engineers with construction planning tasks.
3. There is little computer integration within or to and from the construction planning phase.

The computer tool discussed in this paper is named CONPAS (CONstruction Planning ASsistant). CONPAS suggests alternative construction plans. It uses the experiential and heuristic knowledge of engineers. It uses 3D CAD drawings, so it can play a part in an integrated system for the whole project cycle. The following paragraphs mention how CONPAS resolves those problems.

CONPAS addresses the first problem: there is little knowledge accumulation for the construction planning phase written in manuals. The experiential and heuristic knowledge of engineers can be implemented in CONPAS employing object oriented technology. It can possess deductive knowledge as well. This knowledge allows users to consult CONPAS about the feasibility of their plans. Further, CONPAS can be a sophisticated knowledge accumulation tool for construction planning by incorporating newer knowledge into it. Then, CONPAS will be a substitute for manuals for novice engineers.

CONPAS addresses the second problem: there are few computer tools available in the construction planning phase to help engineers with the early stage of planning tasks. CONPAS accepts the objectives and the environmental condition of a project. It examines actions for the components, the necessary resources, the sequences of activities and the arrangement of resources for the whole project. Those factors form a framework of a construction plan, that explicitly defines also the cost and the duration of the project. Thus, CONPAS can suggest a practical construction plan from the objectives and the environmental condition of a project.

CONPAS can support the third problem: there is little computer integration within, and to and from the construction planning phase. CONPAS has an interface with 3D CAD to obtain input from drawings and to return output to drawings. Since CONPAS has such a function, users operate this tool with drawings in the center of tasks. The architecture for the interface with 3D CAD makes it easy to link CONPAS to other CAD-linked computer tools and makes it possible to realize drawing-centered integration through the work flow of construction projects. Therefore, CONPAS can promote the whole project integration by its function of utilizing 3D CAD drawings.

3. Construction Planning Assistant

3.1 Purpose

The purpose of CONPAS is to assist construction engineers in building construction plans by suggesting the following four items.

1. Zones applicable for a project
2. Required construction equipment to perform the construction procedures
3. Construction procedures, including the sequence of construction activities
4. Fixed equipment lay out and time-dependent equipment lay out

The following section, "Representation", discusses the "zone" in detail.

CONPAS is focused on producing feasible alternatives for executing construction though it is desirable to feed back the result of planning diagnosis to the objectives of the design phase [Fischer]. CONPAS needs the data on the objectives and the environmental condition of a project as the input. It explains a construction plan in a 3D drawing as the output.

CONPAS covers only construction planning for erecting work of structural steel. Erecting-steel work has been selected because it is assumed to be important work, because it is expensive, and because it is a visually noticeable work so that a layman can understand the functions of CONPAS.

Figure 3.1-1 shows the concept of CONPAS.

3.2 Representation

CONPAS is intended to describe the significant types of objects or events involved in a construction execution. CONPAS has several categories of representation for various objects and events in a construction project. Those representations are used as classes in the knowledge-based software CONPAS employs. This section explains the representations of CONPAS.

3.2.1 Essential three representations

There are three representations that CONPAS uses for construction activities: Component, Action, and Resource. The word activity means a segmented work unit of the entire project such as erecting Column_1 or transporting Beam_2. These representations for an activity were defined in PIPPA and successfully adopted to OARPLAN [Marshall, Darwiche]. Darwiche discusses how to represent and to reason about activities. CONPAS applies this representation, though the names of representations are modified to fit them to the purpose and the architecture of CONPAS. The following paragraphs explain those representations.

Component: This representation is equivalent to Object in OARPLAN. It describes every constituent of a building that should be installed or assembled in a construction site. Since CONPAS specializes in erecting structural steel work, the Component represents steel pieces like columns and beams.

Action: This is the equivalent representation to Action in OARPLAN. This describes an action that is performed to achieve an activity. For instance, in the activity "Erect Column_1", an action "Erect_1" is necessary. In CONPAS, three kinds of actions are prepared: Erect, Tighten_Bolts, and Weld.

Resource: This is the equivalent to Resources in OARPLAN. It represents equipment or a tool utilized in executing an activity. Examples of the Resource are, TowerCrane_a, Truck_b, or Welder_c.

This representation enables CONPAS to describe activities and to reason about activities as well. However, OARPLAN defined a project plan as "a list of activities and their sequential relationships" [Darwiche]. In CONPAS, a construction project plan is defined as a set of the objectives, actions, resources, sequence of activities, and arrangement of resources. CONPAS needs additional representations for activities, to enhance the ability so that it examines a sequence of activities and resource arrangement. The following sections elaborate on the additional representations.

3.2.2 Task-Sequence

A series of activities are necessary to complete projects. Planning must create linearized sequence of activities. Although activities have both of preceding and succeeding activities, those activities cannot describe a specified linearized task-sequence. The idea of a linearized sequence is indispensable for examining a construction plan. Figure 3.2-1 shows the difference between a non-linearized sequence and a linearized sequence. The figure shows there are many alternative linearized sequences assumed from a set of activities. A construction plan is examined to find the optimum task-sequence by evaluating those alternatives. The function to create linearized sequences is essential to produce a construction plan. Hence, the representation of Task-Sequence is necessary for CONPAS to make it possible to give flexible alternatives of sequence and a practical construction plan.

3.2.3 Lay-Out

This representation describes both fixed and time-dependent equipment lay-out. In other words, CONPAS is able to describe both where the equipment should be arranged and when the equipment should be arranged. This Lay-Out representation chiefly consists of a resource that is supposed to be laid out and activities to place the resource. Lay-Out representation is essential for CONPAS to produce a realizable construction plan. To program resource arrangements, the computer tool has to describe the location and the times that a resource is installed. However, it is hard for the three representations stated in the foregoing paragraphs to explain such a resource arrangement of intricate task procedure. For instance, it is difficult to explain that equipment is installed at different locations as the task makes progress. An independent representation like Lay-Out realizes a flexible description about lay out even for a complicated task procedure. CONPAS needs the Lay-Out representation to describe resource arrangements in every task procedure of a project.

3.2.4 Zone

Sometimes, construction projects should be divided into several areas to increase work efficiency using concurrent procedures. In addition, the idea is useful to regulate intricate work procedures for grasping the project accurately. The idea to divide a project is termed "zoning."

The idea of "zoning" is discussed in Winstanley's research on aggregation of activities based on zones [Winstanley]. He reviews the utility of the concept of zoning as a capability to represent multiple-levels of aggregation of activities. Moreover, he shows the possibility to represent concurrent tasks and the influences among them. CONPAS uses his idea of zoning. In CONPAS, the use of zones is focused on managing concurrent procedures of tasks rather than on creating realistic work packages.

The Zone representation functions to regulate multiple, complicated series of tasks by dividing the project into several zones that each uses one procedure of tasks.

A zone basically consists of Component, Action, Resource, Task-Sequence, and Lay-Out. Every zone has instances of the five objects. Figure 3.2-2 illustrates the idea of zones in CONPAS. This scheme to separate objects is based on the idea that there should be only one procedure of tasks of a single type of job in one zone. This idea matches a very common manner in the real world to consider how to arrange construction areas. Engineers define zones by identifying concurrent work procedures of single type of job to improve efficiency. Accordingly, it is reasonable to assign one zone to each procedure of tasks expected in a project.

In CONPAS, a zone is treated as the smallest unit for examining a construction plan. The items to examine for producing a construction plan, type of resource or location of equipment, are individually handled in each zone. Hence, zoning must be done prior to the other decision making when a project is presented. Since the main five elements of a project are separately distributed to all the zones, there is no inconvenience to devise a different plan for each zone.

3.2.5 Project

The Project representation includes main the five elements for each zone. An object of this representation includes all of the objects for the same project. The Project representation includes various attributes for describing the entire project such as for the architectural area, for the scale of whole project, or for other environmental specifications.

Those representations discussed in this section are used in the knowledge-based software as shown in the appendix.

3.3 Reasoning

CONPAS suggests alternative construction plans by examining the resource, task-sequence and lay-out for each zone of a project. The zones of a project are suggested by CONPAS as well. This section covers the knowledge implementation in CONPAS.

3.3.1 Experiential knowledge

Construction planning uses experiential knowledge of engineers more than theories on planning. Although there are a few guidelines, theory does not cover detailed decision making. To produce a construction plan, CONPAS suggest four items that are mentioned in the foregoing paragraph: 1. applicable zones, 2. construction procedures, 3. required equipment, 4. both fixed and time-dependent equipment lay out. The knowledge is classified by examining the part to which it applies. The knowledge to examine parts is implemented in CONPAS in the form of rules of the knowledge-base application tool.

Examine zone: Zoning is analyzed based on the environmental condition and the characteristics of the structural steel of a project. The criteria for zoning are classified below.

1. environmental condition
 - a. area of footprint
 - b. ratio of footprint sides
2. characteristics of construction
 - a. stories of structural steel

- b. structure of structural steel
- c. underground part of structural steel

CONPAS deals with a site and a footprint of a building as the environmental conditions of a project. The area of a footprint is one of the criteria to judge the possibility of zoning. If a project has a large building footprint, CONPAS makes a suggestion to divide the construction into zones. Zoning is also grounded on the ratio of footprint sides. If a footprint is thin, rectangular shaped, the possibility of zoning might be high. As well, CONPAS suggests the possibility of zoning for high / low rise part of structural steel, S /SRC part of structural steel, or under / above ground part of structural steel.

Examine resource: CONPAS recognizes the needs of resource types by high level classes, e.g., Cranes, Trucks, Welders. Necessary resource classes are chosen based on the attribute for expected resources of each action that activities in the construction include. Each primitive action in the action class has some resources to anticipate to employ for itself. CONPAS chooses those resources to execute a construction that includes each activity.

The types of crane, e.g., TowerCranes, TruckCranes, CrawlerCranes, are examined in detail. There are criteria for selecting a crane type as follows.

- 1. environmental condition
 - a. side of footprint
 - b. room around footprint
 - c. architectural area
- 2. characteristics of construction
 - a. height of structural steel
 - b. stories of structural steel
 - c. number of steel pieces
 - d. other crane use
- 3. others
 - task-sequence

CONPAS checks the suitability of each crane type referring to the environmental condition, the characteristics of construction of a project, and with the different features of cranes. For instance, a crawler crane needs large room for assembling at the site, and a truck crane has a height limitation for its boom.

Examine task-sequence: CONPAS creates a linearized sequence of activities. However, scores of activities may make almost infinite number of sequences. CONPAS summarizes them in five practical level of sequences, that are specially termed "ordering policies." Figure 3.3-1 shows the five ordering policies. To create a linearized task-sequence, first, CONPAS selects one of ordering policies that is appropriate for a construction; second, it decides the order of activities one by one.

In selecting a ordering policy, CONPAS uses the following criteria.

- 1. environmental condition
 - a. area of footprint
 - b. ratio of footprint sides
 - c. architectural area
 - d. location of entrance
- 2. characteristics of construction
 - a. height of structural steel
 - b. stories of structural steel
 - c. number of steel pieces
 - d. stability of structural steel

3. others resource type

CONPAS selects an ordering policy with evaluations like that the end of erection should be near to the entrance; a horizontal sequence is not applicable exceeding the height limitation; and a vertical sequence is recommended for structural steel with small number of columns due to limited stability.

In creating a linearized sequence, CONPAS uses:

1. physical relationships among steel pieces;
2. stability of structural steel.

The first criterion means that a steel piece with a relationship of, for instance, supporting another piece, must be erected before the piece it supports. The second prevents CONPAS from erecting a column on top of the other column that is not connected to beams.

Examine lay-out: CONPAS examines the lay-out of machinery-equipment-type of resources in a project. The lay-out of a resource is considered as both: fixed resource lay-out and time-dependent resource lay-out. CONPAS suggests the fixed lay-out by selecting a set of x-y-z coordinates through examinations using grids, that are mentioned below in the section 3.3.2. It also suggests the time-dependent lay-out by listing activities that employ the resource.

CONPAS searches for an appropriate location, in x-y-z coordinates, for a site where a resource is to be laid-out. CONPAS first examines whether the location of a resource is inside of the footprint or not. The location of a resource should be adjacent to the work station, namely the footprint of the zone that is examined. Whether the resource location is inside or outside greatly influences the task-sequence or resource capacity. Second, CONPAS searches the desirable location of a resource according to the task-sequence and the capacity of the resource.

A resource location of inside / outside of a footprint is examined based on:

1. resource type;
2. area of footprint.

CONPAS decides a location of inside / outside considering the capacity of the resource type and the area of a footprint. It is sometimes better for a resource with small capacity to be laid-out in a large footprint, and occasionally, the location should be moved as the construction progresses.

A detailed location inside / outside of a footprint is examined based on the following criteria.

- | | |
|----------------------------|--------------------------|
| 1. resource type | capacity |
| 2. task-sequence | ordering policy |
| 3. environmental condition | a. room around footprint |
| | b. location of entrance |

A detailed location is examined so that the capacity of resource is minimal, and it coincides the procedure of activities. Also the location in the end of a construction should be nearer to the entrance.

CONPAS suggests the time-dependent lay-out of a resource by giving a cluster of activities that need the resource. It is examined based on:

- | | |
|------------------|---------------------------|
| 1. resource type | capacity |
| 2. lay-out | fixed lay-out of resource |

Since CONPAS has an idea of linearized sequence of activities for every zone, it is possible to show the time-dependent lay-out by giving each activity needing a resource. Also, it might be possible to measure the actual start and finish time for the use of a resource.

Examining priority: CONPAS makes alternative construction plans by examining resource, task-sequence, and lay-out for each zone. Although human engineers consider those three factors as well, there is no defined procedure for the three factors. The order to examine the three factors, i.e., examining priority, influences a construction plan. Engineers begin their work with a factor that is considered most influential to the plan. Engineers decide each factor's degree of influence by looking at the scale or the characteristics of a project. For instance, when the site is quite small, they examine resource prior to the other two factors because they are worried if large equipment can be used in the site. Then the decision for the resource may have a certain effect on the task-sequence or the lay-out because of the characteristics of the selected resource. The examining priority sometimes changes a construction plan. The engineers experiential knowledge is implemented in CONPAS to decide the examining priority.

To simplify the program, CONPAS is focused on the examining priority of two of the three factors: resource and task-sequence. Each factor's degree of influence is evaluated by following criteria.

1. environmental condition
 - a. area of site
 - b. ratio of site sides
 - c. area of footprint
 - d. ratio of footprint sides
 - e. location of entrance
 - f. width of entrance
2. characteristics of construction
 - a. height of structural steel
 - b. number of steel pieces
 - c. other crane use

CONPAS evaluates examining priority by priority points to each criterion. The values of priority points come from heuristic ideas of engineers. CONPAS suggests that the factor that gets the bigger points should be examined prior to the other factor.

3.3.2 Lay-out examination using grids

Examining lay-out, in other words selecting a set of coordinates, is one of the most difficult tasks for a computer tool because of the almost infinite number of candidates for one solution. Tamaki successfully applied "grids" for his computer tool for designing lay-out of rooms in a building [Tamaki]. He uses grids to model a conceptual lay-out problem as a module of a building. Grids are useful to search a certain set of coordinates since they reduce an infinite number of candidates into finite ones. CONPAS employs the idea of grids in an enhanced form to examine the spatial lay-out of equipment in a site.

A grid is a conceptual object to examine spatial lay-out that has attributes shown in Table 3.3-1. Grids describe the approximate arrangement of a construction site. They are flexible in their size so they can adjust themselves to the scale of a construction or the precision a user desires. CONPAS can transfer the environmental condition of a project from a CAD drawing to grids. Then CONPAS searches an appropriate lay-out of grids, and it returns the lay-out to a CAD drawing in the form of coordinates.

CONPAS examines lay-out by using grids with criteria of adjacency [Tamaki] and distribution. It evaluates the applicability of a lay-out by its adjacency to other objects. For instance, a crane for steel erection should be adjacent to the footprint of the structural steel, or a truck to transport steel pieces should be adjacent to a crane. Meanwhile, an environmental condition of a site or a task-sequence of a construction is concerned with the distribution of resource lay-out. For instance, a task-sequence by x-ordering policy prefers a maldistributed equipment lay-out on the larger-x-coordinates side. For another example, CONPAS select grids in the side of the entrant to lay equipment out according to the location of the entrance because of the convenience of entrance and exit.

The knowledge implemented in CONPAS is shown in the appendix in the form of source code of rules.

3.4 Software Tools

CONPAS is built on a commercial knowledge-based tool, ProKappa¹ with a linkage to AutoCAD². ProKappa and AutoCAD run on SUN work stations with the UNIX operating system.

3.4.1 User interface

CONPAS's user interface is a set of Dialog Boxes and Active Images of ProKappa, and DCL (Dialog Controlling Language) of AutoCAD. CONPAS has several dialog boxes for command menus in ProKappa and dialog boxes with customized command menus for AutoCAD operation. Also, it has two image panels that show users information on the project. Figure 3.4-1 is the overview of the CONPAS's user interface.

On the main menu, in ProKappa, the setting-project sub menu is for registration of a construction project users want to examine. Through the main menu, users can call four sub menus to examine zones, examine resources, examine task-sequences, and examine lay-out. Also, users can make communications with AutoCAD using the reading-drawing and the writing-drawing sub menus to exchange the information on a project. Meanwhile, on the AutoCAD side, users can exchange information, as well. Then, users can get information as drawings through the customized menu.

3.4.2 System interface

CONPAS has a system interface to exchange data with a CAD application. The system interface provides CONPAS with a drawing-oriented style that fits the drawing-centered task flow of a construction project. First, the system interface enables CONPAS to obtain geometrical and mass-property data of a 3D drawing from CAD. Since the information on a construction project is chiefly presented as drawings, this function that interprets geometric models in a drawing to a knowledge-based computer tool in a form of semantic models is a fundamental feature of CONPAS. Second, the system interface provides a function to visualize the process and the result of a construction-plan-production. CONPAS shows a construction plan as a graphic simulation, e.g., illustrating components in a certain zone, simulating the sequence of tasks, or creating an entity of equipment in a certain location. Users can view a CAD drawing rather than results shown in dialog boxes.

¹ An object oriented knowledge engineering environment produced by Intellicorp

² A computer aided design in g tool produced by AutoDesk

Illustrating a construction plan in a CAD drawing is comprehensive, and helps ensure users the feasibility of a plan.

The system interface enables CONPAS to make an instant query & answer about the plan in progress. Users can inquire whether their suggestions are appropriate, and CONPAS answers them by showing diagnoses of their applicability. The query & answer style of operation is effective especially in solving lay out problems that are included in construction planning. After CONPAS examines each of four items: zones, resource, task-sequence, and lay-out, users suggest the type of resource or location of equipment through a 3D CAD drawing. A suggestion from a user is acceptable both before and after CONPAS examines it. CONPAS gives a diagnosis on the applicability of the suggestion, takes it into account in examining the process, then it returns more a practical plan.

Communication between two applications, ProKappa and AutoCAD, is made by UNIX interprocess communication. The device for this interface uses Khedro's research for agent-based technology [Khedro]. The data to exchange between the two are processed into messages formed as lists, then sent to the buffer of a socket. Sockets for communications are prepared for ProKappa and AutoCAD. Messages in a socket can be written and read from both applications through their user interfaces. After getting a message, ProKappa or AutoCAD creates semantic models or geometric models according to the information in the message.

3.5 Test case

This section describes how CONPAS builds a construction plan by presenting two examples. One is a very simple structural steel system prepared for a demonstration. Another is the structural steel of Terman Annex Building that is currently under construction at Stanford. Figure 3.5-1 and 3.5-2 shows the birds' eye views of each case. CONPAS has made realistic plans for each case that are satisfactory from the view point of civil engineers. However, it is necessary to modify to the locations of equipment by suggestions from users because the current version of CONPAS cannot recognize the dimension of equipment.

3.5.1 Demo example

This example building has two stories and 21 pieces of steel: 10 columns and 11 beams. To have few constraints for planning, the example has a rather large site for the size of the building. CONPAS has suggested that it is not necessary to have multiple zones for this project because the scale of the building is small and it has no special characteristics such as underground steel pieces. It advises about the examining priority that resources should be considered prior to the task sequence for the zone. It explains that the priority is due to the small number of steel pieces and the rather slender shape of the footprint. As for the selection of resources, CONPAS has suggested that a truck type of crane should be used for the erection because the scale of the structural steel is small for the use of a tower crane, and because it doesn't have enough room to assemble a crawler crane site. CONPAS has chosen a horizontal task sequence along the x axis from the larger-x-side to the smaller-x-side. A horizontal sequence has been chosen to allow use of a truck crane. Also, the small architectural area and the small footprint area suggest the selection of a horizontal sequence. CONPAS has decided the location of a crane outside of the footprint and in the smaller-x-side of the site. The decision is due to the small footprint that allows a truck crane to erect pieces from the outside, and to the gate location of the site so that it is convenient for entrance and exit. CONPAS points (4.6 , -1) coordinates for the

location of a crane so the location involves the minimum load. It says the minimum load is 17.45 ton meter, and has selected a 25 tons' capacity crane.

3.5.2 Terman Annex building

The Terman Annex Building is a two story building including two lecture theaters. The structural steel of this building consists of about 50 pieces of columns, beams, and trusses. In this test case, some of the steel pieces are omitted so that CONPAS easily handle it. The number of pieces is reduced to 41, 20 columns and 21 beams, and the trusses are treated as irregularly shaped beams.

CONPAS suggests that no zoning is necessary for the Terman project. The Terman project, same as the demo example model, has rather small scale of structural steel and small footprint without any special features. The examining priority is suggested 4 to 3 for the value of sequence priority and resource priority. As shown in Figure 3.5-3, sequence priority is judged from the slender shape of the footprint and small number of steel pieces. Although resource priority is recommended due to the large amount of room around the footprint, the sum of the values for sequence priority is bigger than one for resource priority.

The task sequence for the Terman project is suggested to be in an rx-ordering. It explains either an rx-ordering or an ry-ordering is suitable because the location of the gate of the site is on the smaller-x, smaller-y-side. Further, a task sequence along the x axis is recommended because of the slender shape of the footprint along the x axis (Figure 3.5-3).

CONPAS suggests that the type of resource for the erection should be truck type of cranes. CONPAS explains it has excluded tower cranes because the task sequence suggested for the project, an rx-order, is not preferable for the use of a tower crane. Also, it explains there is little room to assemble a crawler crane in the site to exclude crawler cranes (Figure 3.5-3).

Figure 3.5-4 shows the grid display panel of CONPAS that describes the configuration of the Terman project site. It suggests two locations to lay a truck crane out: one in the middle of the footprint, another on the left side of the footprint. CONPAS means a truck crane should erect the right half of steel from the middle of the footprint. Then, after a crane moves to the left side of the footprint, it should erect the left half. It explains for the lay out inside of the footprint that the footprint is large for the general reach of a truck crane. CONPAS advises the optimum locations according to the minimum load, 2.88 ton meter. Further, it explains that the locations of a crane should be in the left side of the structur because the erecting procedure is from the left to the right (Figure 3.5-3).

To make the plan more practical, a few modifications for the lay out of a crane have been made by suggestions from users. The first location of a crane is shifted to the south almost to touch the property line. That allows workers to prepare other tasks in the left side. The orientation of the crane in the second location is changed 90 degrees so that it doesn't exceed the property line (Figure 3.5-5).

4. Conclusions and Implication

4.1 Contribution of CONPAS

This paper has demonstrated the possibility of application of a knowledge-based computer tool to construction planning. Further, it has proved that CONPAS (CONstruction Planning ASsistant) can assist engineers with construction planning. To accomplish the goal of this research, it: 1. represents a construction plan with five elements, 2. implements experiential knowledge of planning, 3. employs an interactive system interface with 3D CAD.

CONPAS implemented experiential knowledge on structural steel erection work mainly as rules in ProKappa. A rule reflects primitive knowledge of engineers, and rules are clustered in several rule sets as knowledge bases. Using those knowledge bases, CONPAS recommends zones, resources, task sequence, and lay out for steel erection without any big difference from what human engineers do. Though the test cases applied in this research are rather simple and with few constraints, CONPAS has given feasible construction plans for those steel erection projects.

CONPAS employs five major representations to describe a construction plan. It uses two additional representations, Task-Sequence and Lay-Out, plus the three that are adopted from OARPLAN [Darwiche]. It can successfully describe different characteristics of steel erections and various alternatives of a construction plan. By introducing Task-Sequence, CONPAS has described a linearized sequence of activities, that includes the idea of time progress. Consequently, it can give an explanation about the type of resource or the location of equipment by task procedure. Introducing Lay-Out has enabled CONPAS to describe fixed and time-dependent lay out of resources. Since the Lay-Out representation has the attributes related to both location and time, CONPAS has described even a complicated lay out where an equipment moves as the erection proceeds. Thus, a construction plan by CONPAS is accompanied with the idea of time and space, so the plan is comprehensive enough to assist engineers planning.

CONPAS is able to exchange messages with AutoCAD by UNIX interprocess communication. That function has enabled the visualization of the input and the output of the system. A drawing of structural steel with the site condition is smoothly brought in ProKappa as a set of objects with diverse attributes on the geometrical features. A suggested construction plan is described as a drawing in different ways: zones are illustrated by the components included in each zone. Erection procedures are simulated piece by piece, and an equipment newly created in a drawing is set in the appropriate location and time. All of those illustrations in a CAD drawing are much more understandable than the output by script or mere coordinates. The easy verification of a suggested plan by such visuals ensures more productivity of planning task utilizing a computer tool.

4.2 Implication for the future

CONPAS is a prototype computer tool for the assistance in construction planning. Although CONPAS draws several ways to improve the productivity of the planning phase, and it has made achievements, this computer tool must be extended in some respects.

Although two test cases have been successfully examined by CONPAS, it has knowledge only to deal with construction projects with few constraints. With the current version of CONPAS, it is hard to suggest an appropriate plan for complicated cases for which engineers really need some assistance. To enhance such an ability of CONPAS, more knowledge of human experts needs to be implemented.

Knowledge for the erection from the road in front of a site, or knowledge for the erection of underground floors from a jetty is at least necessary to cope with more complicated constructions. Extension of CONPAS for other types of work, such as concrete work or temporary work, would be the next step.

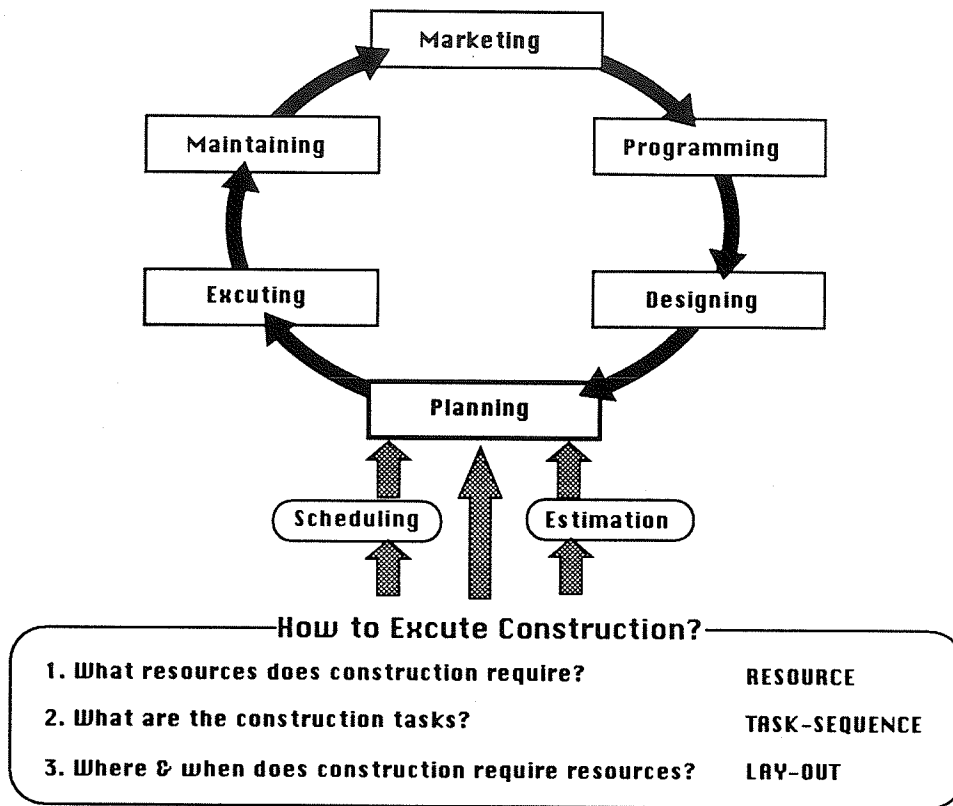
Also, the user interface related to CAD should be improved. The current version of CONPAS allows users to make suggestions through a CAD drawing in a limited manner. Users have to input a type of resource or a location of equipment by manually typing scripts in some dialog boxes. CONPAS needs the easier and the more flexible user interface with full advantage of a drawing. Since the interactive operation between a computer tool and a human is an effective way of polishing a construction plan, improving the user interface ensures more productive planning.

CONPAS has exemplified the possibility to build a computer tool assisting engineers in construction planning. It also realizes computer integration with the "drawing centered" vision. The endeavor to improve this computer tool should be continued so it matches the practical use of construction planning.

References

- Darwiche, A., Levitt, R. E. and Hayes-Roth, B.
"OARPLAN: Generating Project Plans in a Blackboard System by Reasoning about Objects, Actions and Resources" CIFE (Center for Integrated Facility Engineering), Technical Report No. 2, February 1989, 11p
- Fischer, M. A.
"Constructibility Input to Preliminary Design of Reinforced Concrete Structures" CIFE (Center for Integrated Facility Engineering), Technical Report No. 64, July 1991, pp. 2-6
- Fruchter, R., Mark, J. C., Krawinkler, H., Kunz, J. C., and Teicholz, P.
"Interdisciplinary Communication of Design Critique in the Conceptual Design Stage" ASCE Fifth International Conference in Computing in Civil Engineering, June 1993, 2p
- Khedro, T., Teicholz, P., and Genesereth, M. R.
"Agent-Based Technology for Facility Design Software Integration" ASCE Fifth International Conference in Computing in Civil Engineering, June 1993
- Marshall, G., Barber, T. J., and Boardman, J. T.
"A Methodology for Modeling a Project Management Control Environment" IEEE Proceedings 134 (Pt. D, 4), 1987
- Oberlender
"Project Management for Engineering and Construction" McGraw-Hill, Inc., New York NY, 1993, 5p
- Shokoku-sha
"Sekou Keikaku Gaido Bukku (Kouji-hen 1)" Shokoku-sha, Tokyo Japan, June 1985, 7p
- Tamaki, M., Kunz, J. C.
"The System for Knowledge-based Layout (SKY) Project" CIFE (Center for Integrated Facility Engineering), Technical Report No. 76, October 1992
- Winstanley, G, Hoshi, K
"Model-Based Planning Utilizing Activity Aggregation Based on Zones" CIFE (Center for Integrated Facility Engineering), Technical Report No. 69, July 1992

Figure 1.1-1



The tasks of a construction project flow circularly. It begins with the marketing phase and goes around to the maintaining phase. This paper is focused on ways to execute a construction by examining resources, task sequence, and lay out. However, for a construction plan, scheduling and cost estimation are very essential factors, considering how to execute a construction is crucial to realize scheduling and estimation.

(Figure 3.1-1 on the next page) To assist engineers with construction plans, CONPAS uses architectural and structural design as its input from the design phase, and gives construction plans to the execution phase. CONPAS mainly deals with its input and output as electronic data through 3D CAD drawings. Also, users view 3D CAD drawings in the operations. The knowledge based part of CONPAS suggests a construction plan by examining Component, Action, Resource, Task-Sequence, and Lay-Out.

Figure 3.1-1

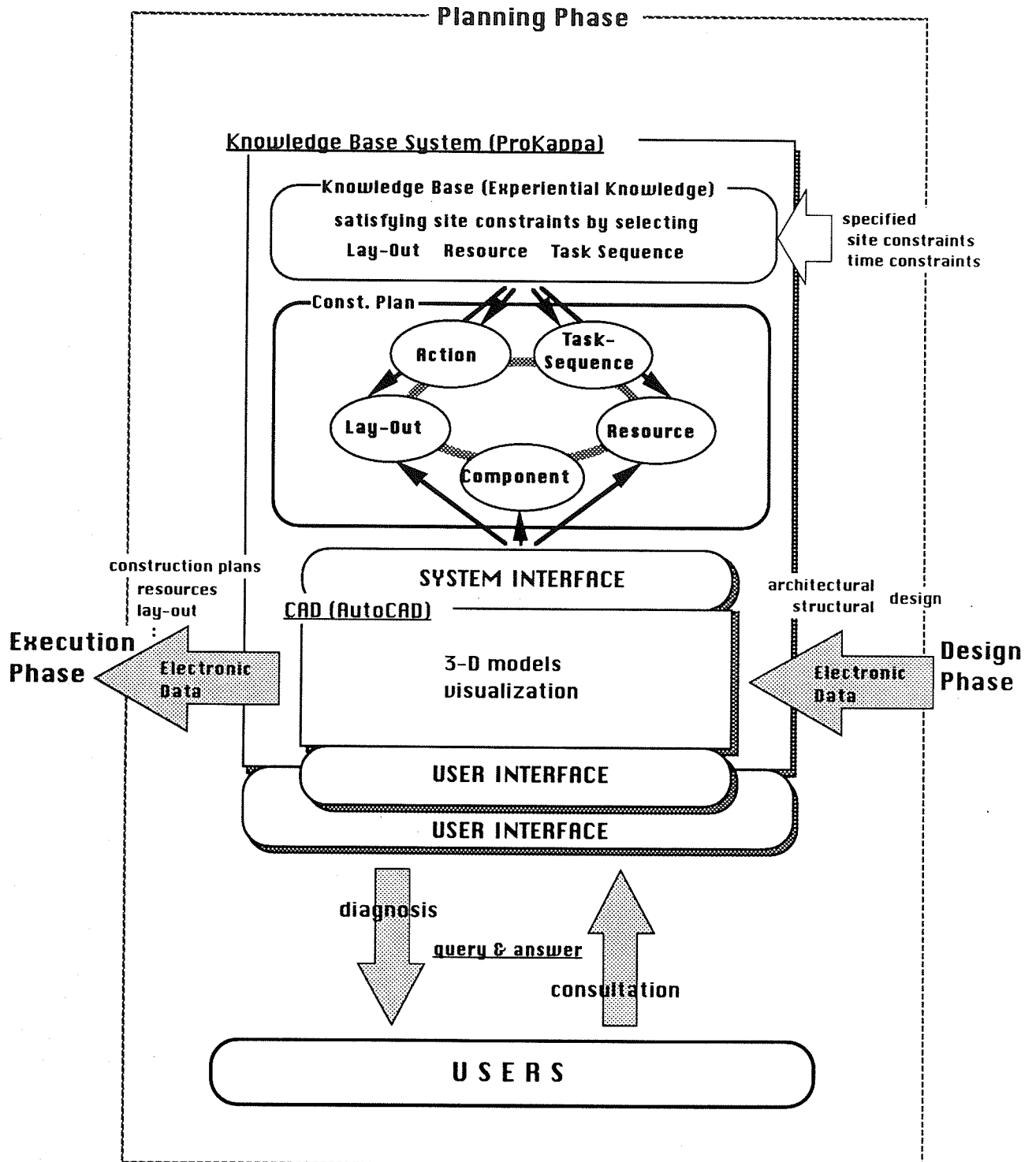
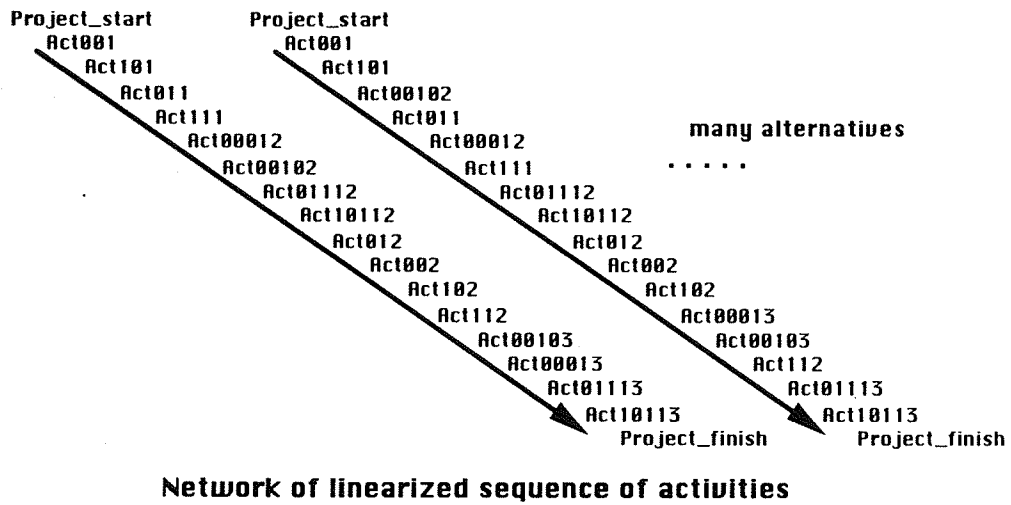
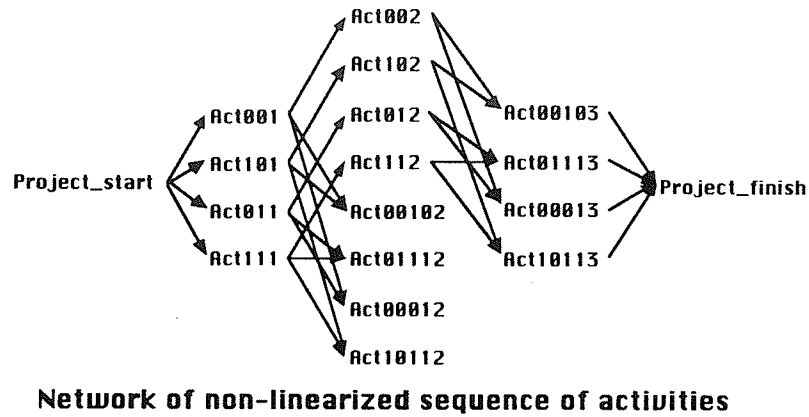
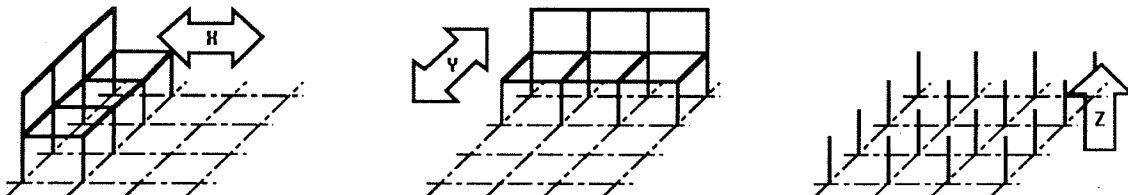


Figure 3.2-1



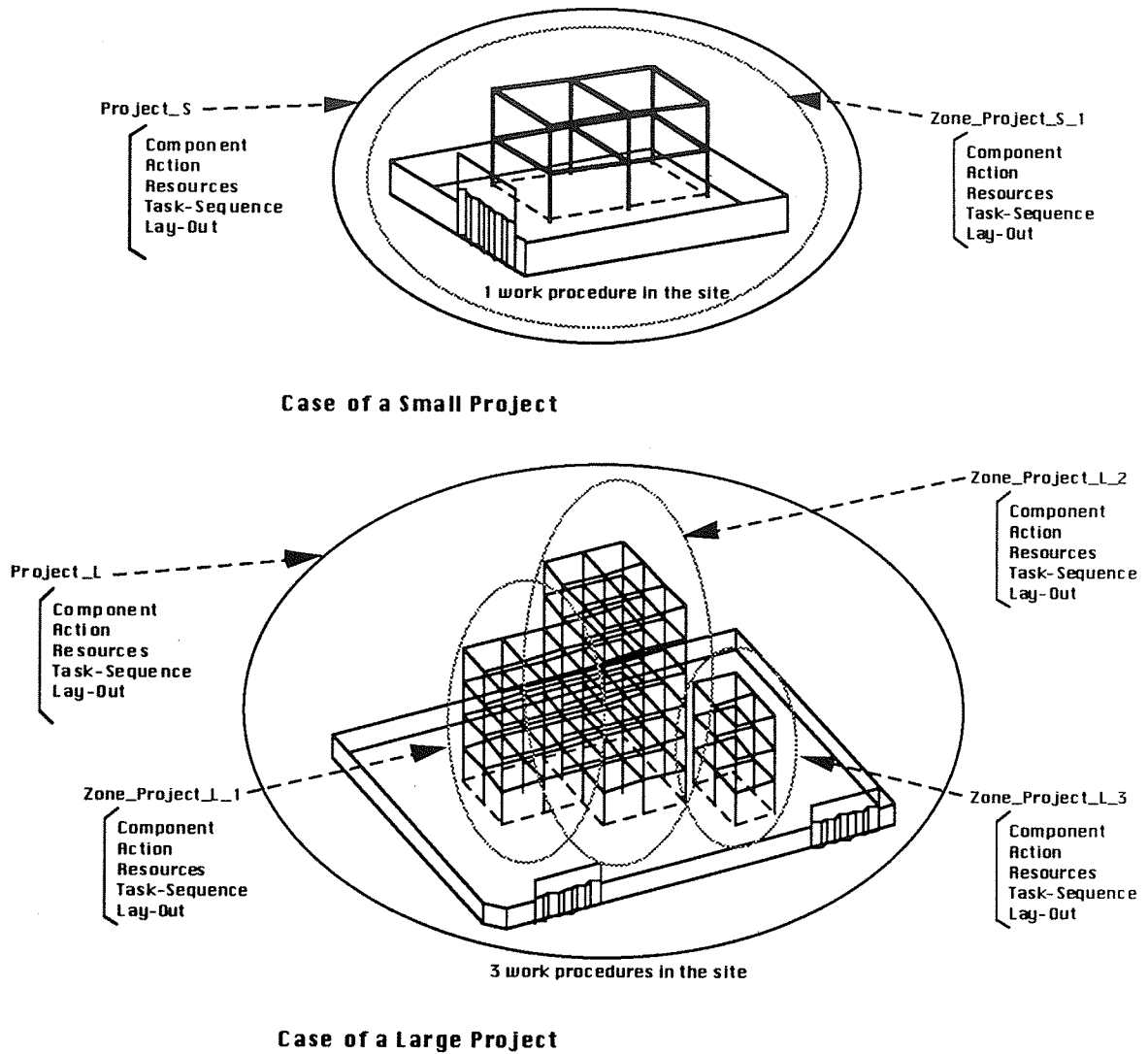
Network of linearized sequence of activities can describe many alternative task sequences assumed from a set of activities. CONPAS can suggest a practical task sequence by evaluating these alternatives.

Figure 3.3-1



A structural steel erection can usually proceed in an x-direction, a reverse-x-direction, a y-direction, a reverse-y-direction, or in a z-direction. A reverse-direction means a task sequence from the side of larger coordinates to the side of smaller ones.

Figure 3.2-2



A zone consists of Component, Action, Resource, Task-Sequence, and Lay-Out. Since CONPAS arranges zones for a project so that each zone includes only one task sequence, there is at least one zone in a project. After CONPAS arranges zones, it suggests construction plans for each zone separately.

Figure 3.4-1

COMPAS uses Dialog Box and Active Image of ProKappa, and Dialog Controlling Language of AutoCAD. The figure shows the hierarchy of menus. There are two systems of menus: one is for the knowledge based part (ProKappa), another is for the CAD part (AutoCAD). These two systems are connected to each other by the UNIX interprocess communication. The communication is processed through Read / Write Drawing menu in ProKappa and Set Communication, Send Model, Parse Message, and Make Suggestion menu in AutoCAD. The menus in the knowledge base part work for examining or evaluating construction plans. While, the menus in AutoCAD work for the data input and output through 3D CAD models.

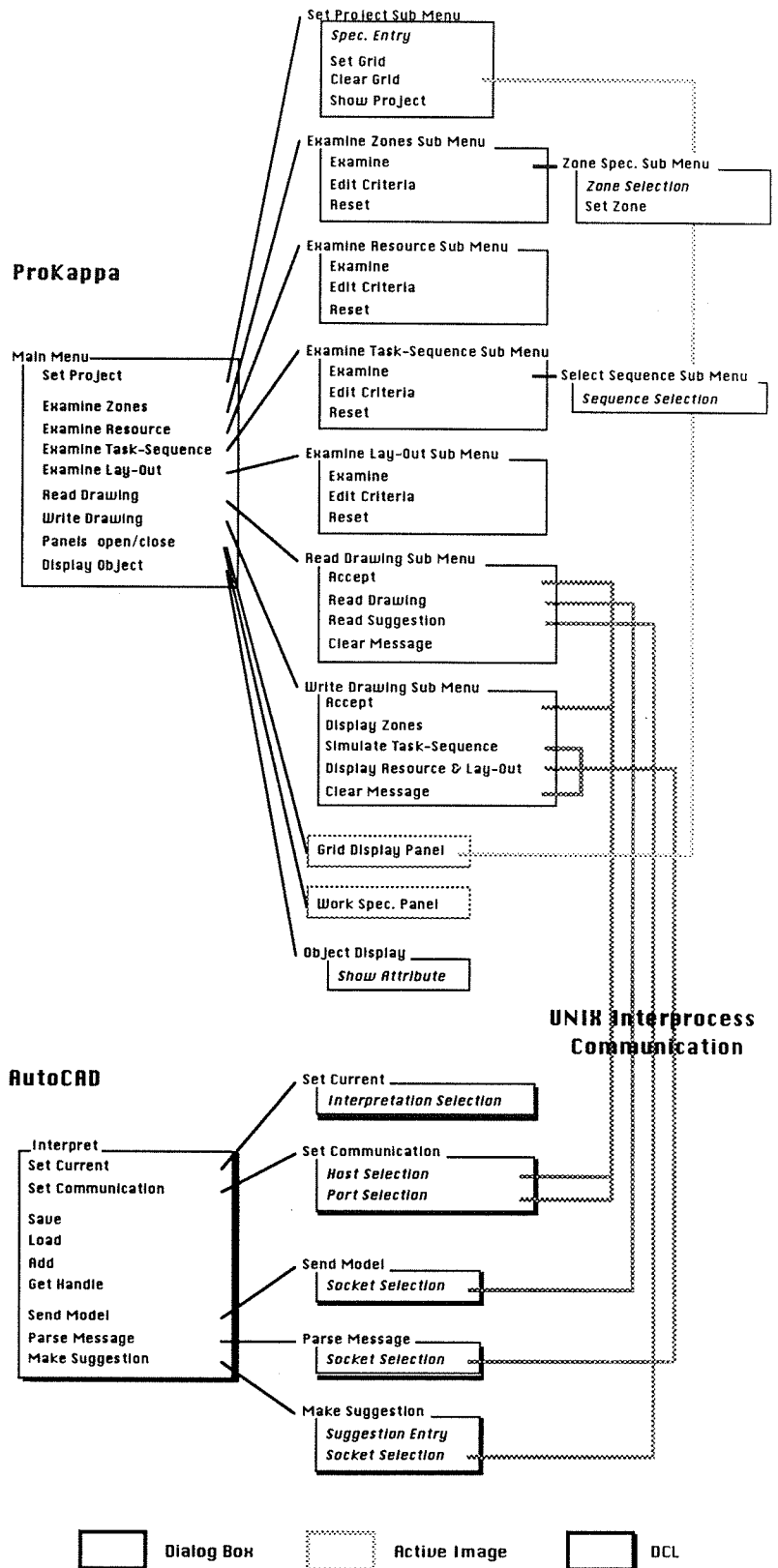


Figure 3.5-1

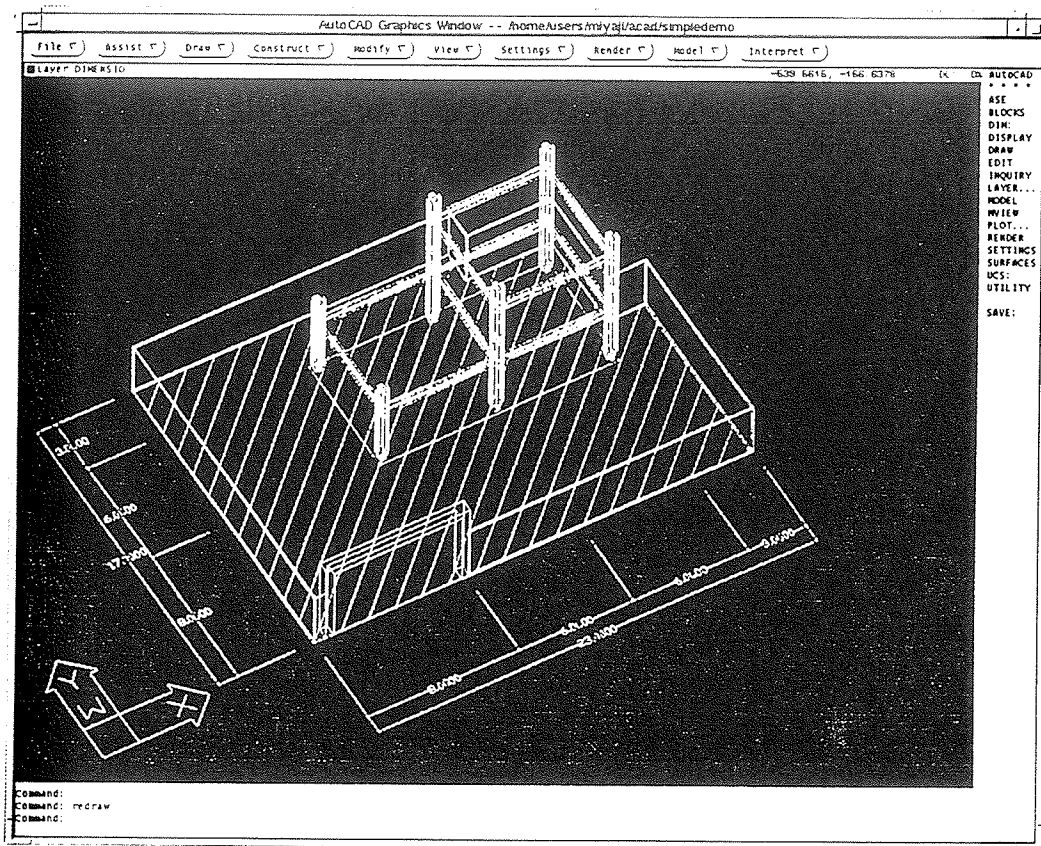


Figure 3.5-2

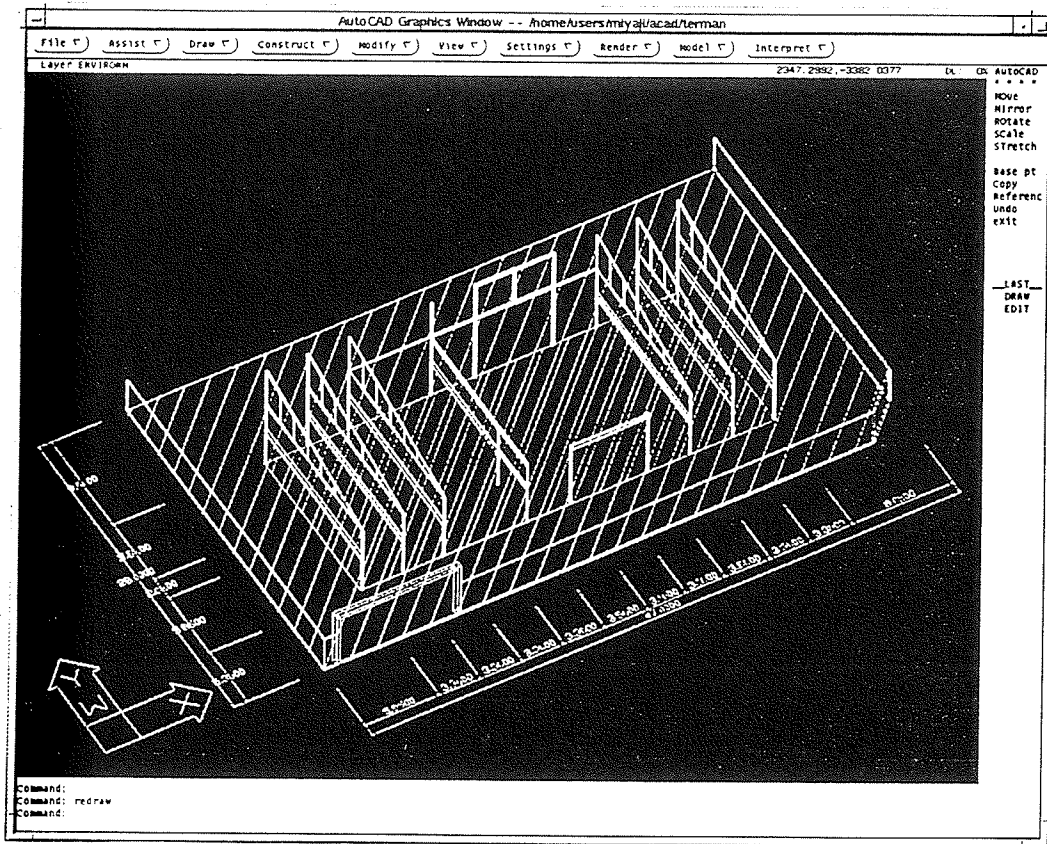


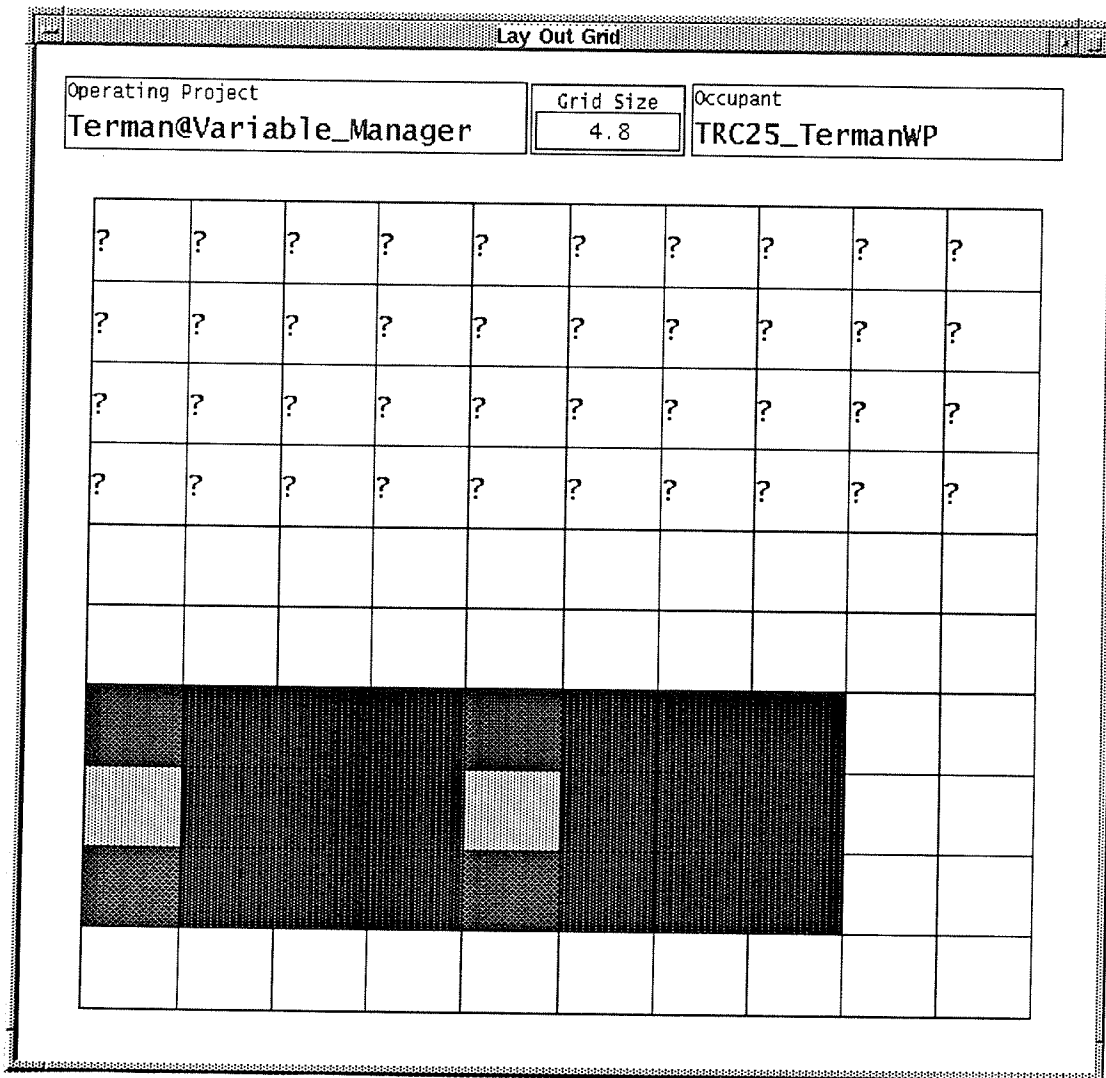
Figure 3.5-3

Work Spec. Panel	
Project Name TermanVariable_Manager	Zone Name TermanWP@Lay_Out
Site Name SitesDBF7@Resources	Footprint Name FpTermanWP@Resources
Footprint Name FootprintsDC2E@Resources	
Architectural Area 760	
Gate Name Gates4B17@Resources	Resource Priority 3
	Sequence Priority 4
S Column 20	S Beams 21
SRC Column 0	SRC Beam 0
Max X 33.1	Min X -2e-06
Max Y 13.8	Min Y -0.075
Max Z 9.75	Min Z 0
Posi. Max X 11	Posi. Min X 1
Posi. Max Y 4	Posi. Min Y 1
Posi. Max Z 4	Posi. Min Z 1
Zoning Possibility ?	Erection Sequence Type RX
Zoning way ?	Resource Option ?
	Crane Type TRC25_TermanWPGResources
Update Project ?	Update Zone ?
Expane Project ?	Explane Zone ?

Explanation Panel	
Explanation/Reasons	<p>Multiple Lay Out: LayOut_TermanWP_2</p> <p>Laying- Out Small X: According to the rx-order task procedure</p> <p>Laying- Out inside: The area of the footprint is large for a TruckCranes Max load is 2.8798</p> <p>Exclude Tower Cranes: The task sequence has already been chosen, which does't prefer tower cranes.</p> <p>Exclude Crawler Cranes: There is little room around the footprint of this zone.</p> <p>TruckCranes Class is selected for this zone.</p> <p>Horizontal Sequence: The architectural area of this zone is small.</p> <p>RY Sequence: The location of the gate is biased preferring RY sequence.</p> <p>RX Sequence: The location of the gate is biased preferring RX sequence.</p> <p>X Axis Sequence: This zone is rather long along the X axis.</p> <p>Sequence Priority: The number of the pieces in this zone is rather small.</p> <p>Sequence Priority: The shape of the footprint is slendar figured.</p> <p>Resource Priority: There is much room around the footprint in the siteground.</p>
	<p>QUIT</p>

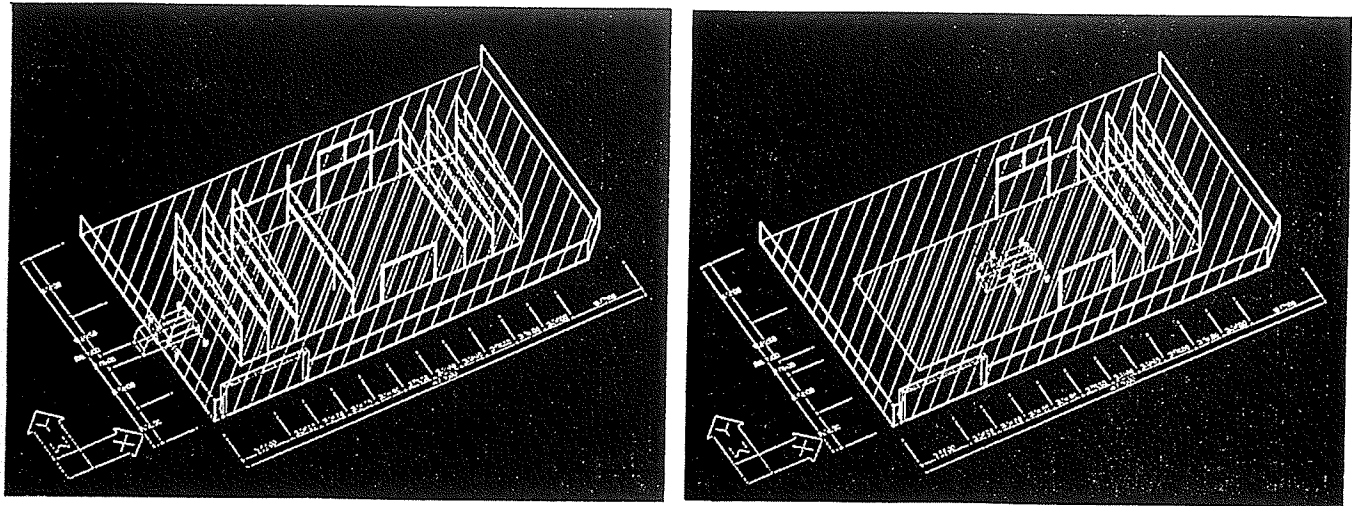
The Work Spec. Panel on the left side shows the detail specifications of the currently working project and zone including resources and task sequence they need. CONPAS explains each suggestion it makes for zoning and selections of resources, task sequence, and lay out. It accumulates all the explanations in the Explanation Panel shown on the right side. Users can get comments at any time for each decision by CONPAS.

Figure 3.5-4

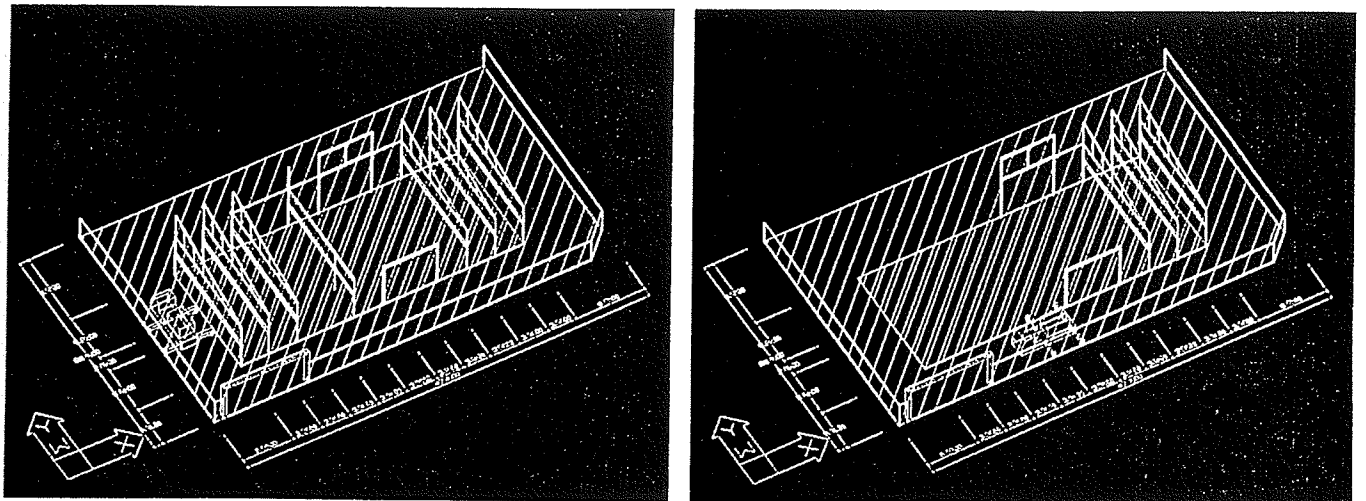
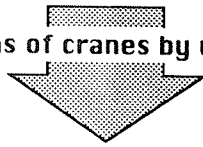


In the panel, white grids represent the site, the darkest grids for the footprint, the second darkest grids for the candidates for a crane's locations, and the light gray grids for the optimum locations for a crane. Each side of a grid is currently set 4.8 meters.

Figure 3.5-5



Change the locations of cranes by users' suggestions



Two locations of a crane are adjusted by users' suggestions. The location in the middle of the footprint is moved to the south so as not to interfere the work in the left side of the building. The orientation of a crane on the left end is changed 90 degrees so that it is inside the site.

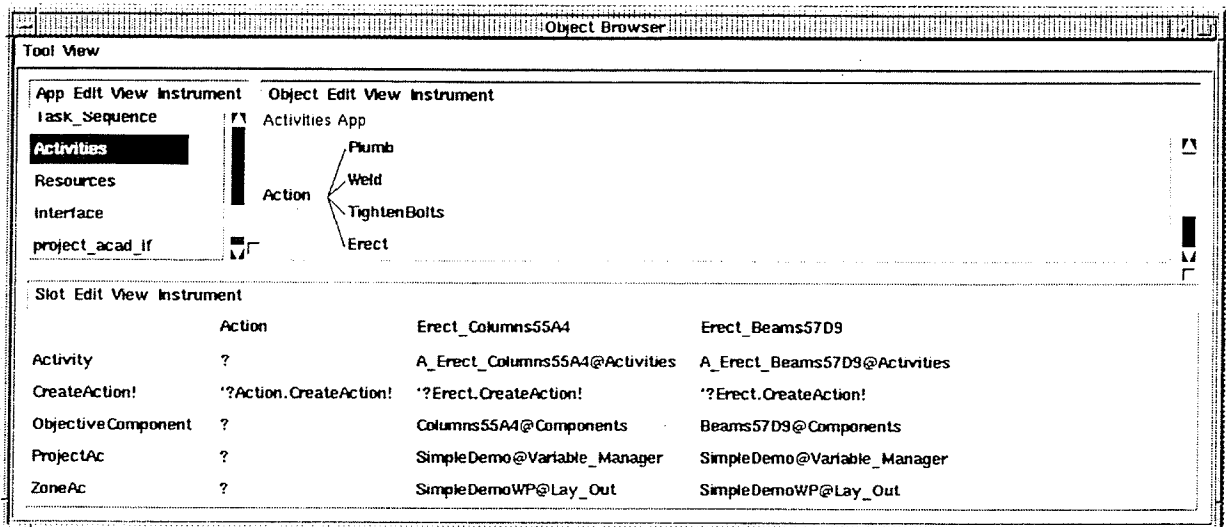
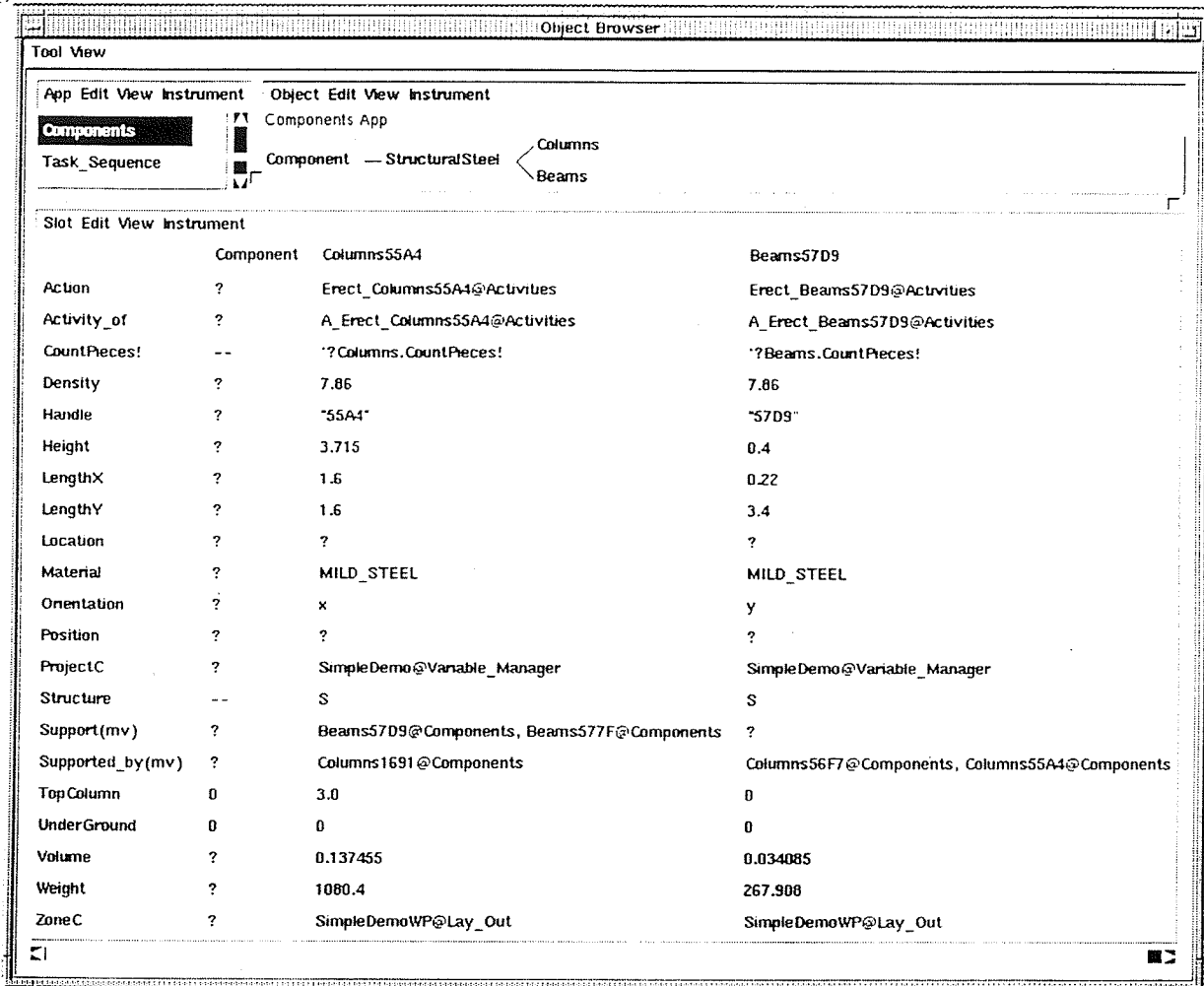
Table 3.3-1

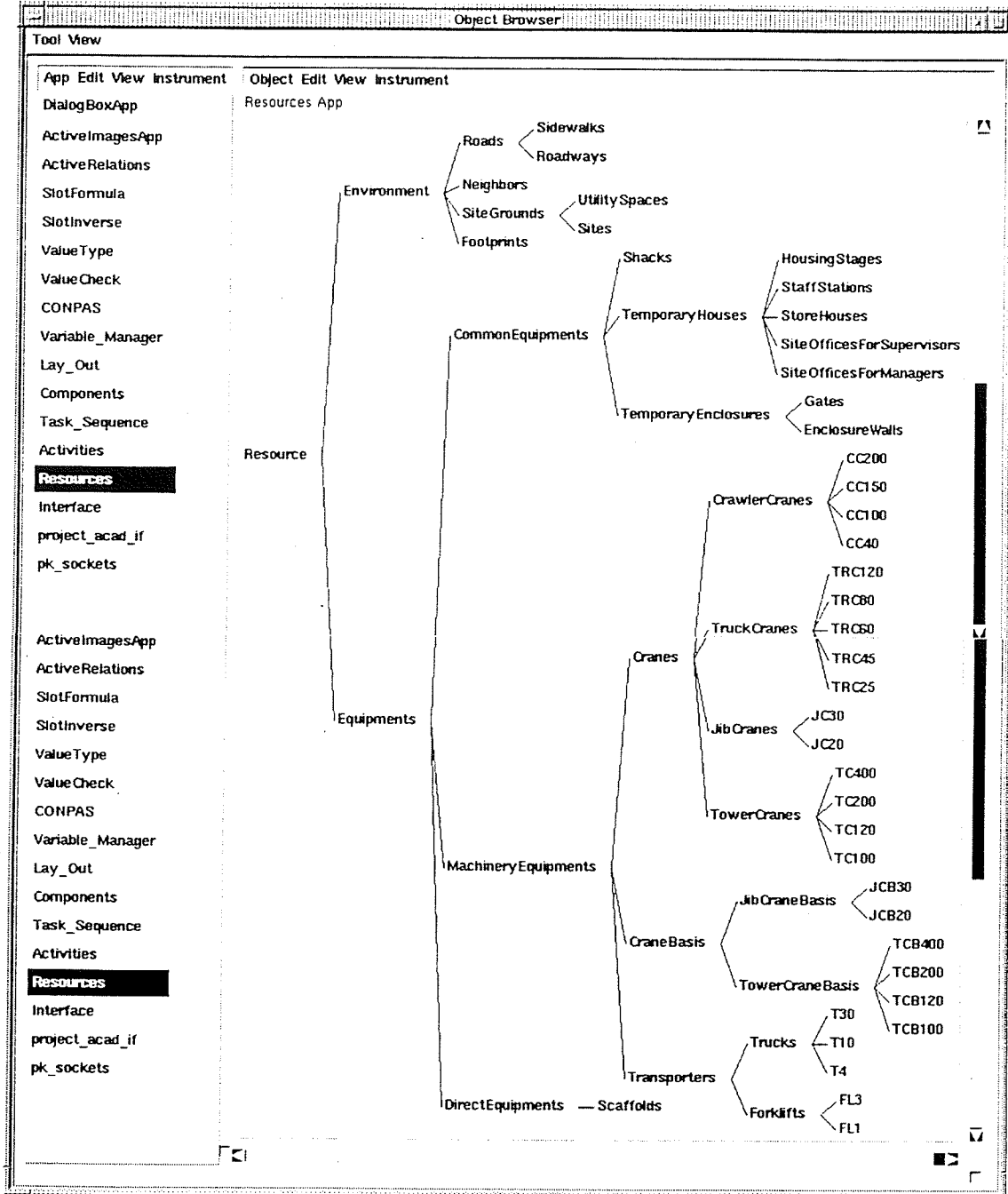
ATTRIBUTE	MEANING	METHOD	MEANING
Center_X	X coordinate of center point of grid	Adjust!	adjust the origin of coordinates
Center_Y	Y coordinate of center point of grid	Clear!	erase the occupant of grid
East	adjacent grid of the east side	FindCrossPoint!	judge whether grid is included in a polygon
Image	image object of grid	FindNearestGrid!	find the nearest grid to a given coordinates
InitialOccupant	resource that initially occupies grid	FindNeighbors!	enumerate the adjacent grids of grid
LayOut(mv)	LayOut object in that grid is included	GiveScale!	set the length of grid
North	adjacent grid of the north side	Occupy!	assign a resource to grid
NorthEast	adjacent grid of the north-east side	ResetGrid!	reset the occupant of grid to the initial state
NortheWest	adjacent grid of the north-west side	ShowEnvironment!	show the current occupant (environmental resource) of grid for a given project
Occupant	resource that currently occupies grid		
Side	length of the side of grid by meter	ShowEquipmentLocation!	show the current occupant (equipment) of grid for a given project
South	adjacent grid of the south side		
SouthEast	adjacent grid of the south-east side		
SouthWest	adjacent grid of the south-west side		
West	adjacent grid of the west side		
X	grid number for X coordinate		
Y	grid number for Y coordinate		

Attributes of an object are classified into simple attributes and methods. Simple attributes of a grid mainly contain the information of the position or of the positional relationships with other grids. Methods of a grid work for shifting views or for judging whether it satisfies a certain requirement.

Appendices

A. Representation in CONPAS





Slot Edit View Instrument

Environment	SitesDBF7	Equipments	TRC25_TermanWP
Area ?	1311.5	--	--
Assembly --	--	--	?
BoomLength --	--	--	?
Capacity --	--	--	?
ClassLevel --	--	?	3
Color ?	White	?	Red
Comp Ground Range! '?Environment.Comp GroundRange!	'?Environment.Comp GroundRange!	--	--
CoordinatesList ?	(-5.95, -5.3 ...)	--	--
Create Resource Object!	--	'?Equipments.Create Resource Object!	'?TruckCranes.Create Resource Object!
EquipedArea --	--	?	?
GeneralLoad --	--	--	?
GeneralRadius --	--	--	10
GiveLocation! --	--	'?Equipments.Give Location!	'?Equipments.Give Location!
Give Orientation! --	--	'?Equipments.Give Orientation!	'?Equipments.Give Orientation!
Handle ?	"DBF7"	?	?
In/Out/AreaLimit --	--	--	200

Slot Edit View Instrument

JibLength --	--	--	?
Location ?	?	?	?
MaxLoad --	--	--	25.0
MaxMoment --	--	--	25.0
MaxRadius --	--	--	20.0
NoofGrids ?	?	?	?
Option --	--	?	?
Orientation ?	?	?	0.001
ProjectR ?	Terman@Variable_Manager	?	Terman@Variable_Manager
SelectOptCranes!	--	--	'?Cranes.SelectOptCranes!
TypeofGroup ?	?	?	?
XMax ?	41.05	--	--
XMin ?	-5.95	--	--
YMax ?	22.7	--	--
YMin ?	-5.3	--	--
ZoneR ?	TermanWP@Lay_Out	?	TermanWP@Lay_Out

Object Browser

Tool View

App Edit View Instrument Object Edit View Instrument

Task_Sequence Task_Sequence App
 TaskSequence — ErectionSequence

Slot Edit View Instrument

	TaskSequence	ErectionSequence_TermanWP
Activity(mv)	?	Terman_Finish@Activities, Terman_Start@Activities ...
CreateSequence!	--	?ErectionSequence.CreateSequence!
CreateSequenceObject!	?TaskSequence.CreateSequenceObject!	?TaskSequence.CreateSequenceObject!
HorizontalSequence	Yes	Yes
OrderingPolicy(mv)	?	rx
ProjectT	?	Terman@Variable_Manager
SelectOrderingPolicy!	--	?ErectionSequence.SelectOrderingPolicy!
Sequence(mv)	?	Terman_Finish@Activities, A_Erect_Beams8CCC@Activi...
VerticalSequence	Yes	No
ZoneT	?	TermanWP@Lay_Out

Object Browser

Tool View

App Edit View Instrument Object Edit View Instrument

Variable_Manager Lay_Out App
 Lay_Out LayOut

Slot Edit View Instrument

	LayOut	LayOut_TermanWP_1
Activity(mv)		A_Erect_Columns4209@Activities, A_Erect_Columns41B...
CalculateMoment!	?LayOut.CalculateMoment!	?LayOut.CalculateMoment!
CreateLayOutObject!	?LayOutL.CreateLayOutObject!	?LayOutL.CreateLayOutObject!
ExaminingStatus	not_examined	examined
Location	?	inside
Locations(mv)	?	E3@Variable_Manager
Orientation	?	0.001
ProjectL	?	Terman@Variable_Manager
Resource	?	TRC25_TermanWP
SupplementActivity!	?LayOut.SupplementActivity!	?LayOut.SupplementActivity!
ZoneL	?	TermanWP@Lay_Out

Tool View		
Object Browser		
Slot Edit View Instrument		
Action(mv)	?	Terman
Activity(mv)	?	Erect_Beams4220@Activities, Erect_Beams4221@Activi...
ArchitecturalArea	?	A_Erect_Columns41E5@Activities, A_Erect_Columns41F...
Component(mv)	?	750.0
CriteriaImportanceR	0	BeamsC6D8@Components, Beams8F70@Components ...
CriteriaImportanceT	0	0
Enclosure	?	EnclosureWallsB843@Resources
Explanation(mv)	?	?
Footprint	?	FootprintsDC2E@Resources
FootprintCoverage	?	?
Footprints(mv)	?	FpTermanWP@Resources, FootprintsDC2E@Resources
GateName(mv)	?	Gates4817@Resources
GridSize	?	4.8
LayOut(mv)	?	LayOut_TermanWP_2@Lay_Out, LayOut_TermanWP_1@Lay_O...
LeftLowX	?	-5.55
LeftLowY	?	-5.3
LongSide	?	33.1
Max_X	-9999	33.1
Max_Y	-9999	13.775
Max_Z	-9999	9.75
Min_X	9999	-2e-06
Min_Y	9999	-0.075
Min_Z	9999	0.0
OtherCraneUse(mv)	?	?
Slot Edit View Instrument		
PositionMax_X	-99	11.0
PositionMax_Y	-99	4.0
PositionMax_Z	-99	4.0
PositionMin_X	99	1.0
PositionMin_Y	99	1.0
PositionMin_Z	99	1.0
ResetActivity!	*?WorkSpec.ResetActivity!	*?WorkSpec.ResetActivity!
ResetComponent!	*?WorkSpec.ResetComponent!	*?WorkSpec.ResetComponent!
ResetLayOut!	*?WorkSpec.ResetLayOut!	*?WorkSpec.ResetLayOut!
ResetResource!	*?WorkSpec.ResetResource!	*?WorkSpec.ResetResource!
ResetTaskSequence!	*?WorkSpec.ResetTaskSequence!	*?WorkSpec.ResetTaskSequence!
Resource(mv)	?	TRC25_TermanWP@Resources, TruckCranes_TermanWP@Res...
SBeamNo	?	21
SColumnNo	?	20
ShortSide	?	13.85
Site	?	SitesDBF7@Resources
SRCBeamNo	?	0
SRCColumnNo	?	0
TaskSequence(mv)	?	ErectionSequence_TermanWP@Task_Sequence
TopStories(mv)	?	?
XCoordUnit	?	3.3
YCoordUnit	?	4.5
ZCoordUnit	?	3.25
Zones(mv)	?	TermanWP@Lay_Out
Zoning	?	?
ZoningWay(mv)	?	?

Object Browser		
Tool View		
Slot	Edit	View Instrument
Action(mv)	?	TermanWP Erect_BeamsC5D8@Activities, Erect_Beams8F70@Activi...
Activity(mv)	?	Terman_Finish@Activities, Terman_Start@Activities ...
ArchitecturalArea	?	750.0
Component(mv)	?	BeamsC5D8@Components, Beams8F70@Components ...
CreateStoryZone!	*?Zone.CreateStoryZone!	*?Zone.CreateStoryZone!
CreateStructureZone!	*?Zone.CreateStructureZone!	*?Zone.CreateStructureZone!
CreateUnderGroundZone!	*?Zone.CreateUnderGroundZone!	*?Zone.CreateUnderGroundZone!
CreateWholeProjectZone!	*?Zone.CreateWholeProjectZone!	*?Zone.CreateWholeProjectZone!
CreateXYZZone!	*?Zone.CreateXYZZone!	*?Zone.CreateXYZZone!
CreateZoneFootprint!	*?Zone.CreateZoneFootprint!	*?Zone.CreateZoneFootprint!
CriteriaImportanceR	0	3
CriteriaImportanceT	0	4
Enclosure	?	?
Explanation(mv)	?	*Laying-Out Inside: The area of the footprint is l...
Footprint	?	FpTermanWP@Resources
FootprintCoverage	?	?
GateName(mv)	?	Gates4B17@Resources
InheritAttributes!	*?Zone.InheritAttributes!	*?Zone.InheritAttributes!
LayOut(mv)	?	LayOut_TermanWP_2@Lay_Out, LayOut_TermanWP_1@Lay_0...
LongSide	?	33.1
Max_X	-99999	33.1
Max_Y	-99999	13.775
Max_Z	-99999	9.75
Min_X	99999	-2e-06
Min_Y	99999	-0.075
Min_Z	99999	0.0
Slot	Edit	View Instrument
OtherCraneUse(mv)	?	?
PositionMax_X	-99	11.0
PositionMax_Y	-99	4.0
PositionMax_Z	-99	4.0
PositionMin_X	99	1.0
PositionMin_Y	99	1.0
PositionMin_Z	99	1.0
ProjectZ	?	Terman@Variable_Manager
ResetActivity!	*?WorkSpec.ResetActivity!	*?WorkSpec.ResetActivity!
ResetComponent!	*?WorkSpec.ResetComponent!	*?WorkSpec.ResetComponent!
ResetLayOut!	*?WorkSpec.ResetLayOut!	*?WorkSpec.ResetLayOut!
ResetResource!	*?WorkSpec.ResetResource!	*?WorkSpec.ResetResource!
ResetTaskSequence!	*?WorkSpec.ResetTaskSequence!	*?WorkSpec.ResetTaskSequence!
Resource(mv)	?	TRC25_TermanWP@Resources, TruckCranes_TermanWP@Res...
SBeamNo	?	21
SColumnNo	?	20
ShortSide	?	13.85
Site	?	SitesDBF7@Resources
SRCBeamNo	?	0
SRCColumnNo	?	0
SupplementComponent!	*?Zone.SupplementComponent!	*?Zone.SupplementComponent!
SupplementStartFinish!	*?Zone.SupplementStartFinish!	*?Zone.SupplementStartFinish!
TaskSequence(mv)	?	ErectionSequence_TermanWP@Task_Sequence
TopStories(mv)	?	4.0, 3.0
XCoordUnit	?	3.3
YCoordUnit	?	4.5
ZCoordUnit	?	3.25
ZoningWay	?	?

B. Source code of rules

Rules for zoning

```
ruleset ZoningRules {}

fcrule XYRule in ZoningRules
{if:
    XY1.ShortSide <= CurrentWork.OperatingProject.ShortSide;
    ?ratio = ConvertToFloat(CurrentWork.OperatingProject.LongSide)/
            ConvertToFloat(CurrentWork.OperatingProject.ShortSide);
    ?ratio >= XY1.SideRatio;
    CurrentWork.OperatingProject.Footprint.Area > XY1.FpAreaLimit;
then:
    CurrentWork.OperatingProject.Zoning = possible;
    CurrentWork.OperatingProject.ZoningWay += XY;
    Symbol.MultiSymbol_1 += ZoningXY.Text;
}

fcrule StoryRule in ZoningRules
{if:
    ?storylist = all CurrentWork.OperatingProject.TopStories;
    ListLength(?storylist) >= 2;
    for ?piece = instanceof Columns;
    do {?piece.ProjectC == CurrentWork.OperatingProject;
        ?piece.TopColumn != 0;
        collect ?piece into ?toplist;}
    for ?story inlist ?storylist;
    always {for ?topcolumn inlist ?toplist;
        do {?topcolumn.TopColumn == ?story;
            collect ?topcolumn into ?list;}
        ?no = ConvertToFloat(ListLength(?list));
        ?tno = ConvertToFloat(ListLength(?toplist));
        100*?no/?tno >= Story1.ColumnRatio;}
    ?no >= Story1.NoOfColumn;
then:
    CurrentWork.OperatingProject.Zoning = possible;
    CurrentWork.OperatingProject.ZoningWay += Story;
    Symbol.MultiSymbol_1 += ZoningStory.Text;
}

fcrule StructureRule in ZoningRules
{if:
    CurrentWork.OperatingProject.ShortSide*CurrentWork.OperatingProject.
        LongSide >= Str1.FpAreaLimit;
    ?No1 = ConvertToFloat(CurrentWork.OperatingProject.SBeamNo);
    ?No2 = ConvertToFloat(CurrentWork.OperatingProject.SColumnNo);
    ?No3 = ConvertToFloat(CurrentWork.OperatingProject.SRCBeamNo);
    ?No4 = ConvertToFloat(CurrentWork.OperatingProject.SRCColumnNo);
    ?ratio = (?No3+?No4)/(?No1+?No2+?No3+?No4)*100;
    ?ratio >= Str1.StrRatio;
    ?ratio <= 100-Str1.StrRatio;
then:
    CurrentWork.OperatingProject.Zoning = possible;
    CurrentWork.OperatingProject.ZoningWay += Structure;
    Symbol.MultiSymbol_1 += ZoningStructure.Text;
}

fcrule UnderGroundRule in ZoningRules
{if:
    ?piece == instanceof StructuralSteel;
    ?piece.ProjectC == CurrentWork.OperatingProject;
    for?piece.UnderGround != 0;
    do collect ?piece into ?piecelist;
    ConvertToFloat(ListLength(?piecelist)) > 3;
then:
    CurrentWork.OperatingProject.Zoning = possible;
    CurrentWork.OperatingProject.ZoningWay += UnderGround;
    Symbol.MultiSymbol_1 += ZoningUG.Text;
}

fcrule ScaleRule in ZoningRules
{if:
    CurrentWork.OperatingProject.ShortSide > Scale1.ShortSide;
    CurrentWork.OperatingProject.ArchitecturalArea > Scale1.ArchAreaLimit;
then:
    CurrentWork.OperatingProject.Zoning = possible;
    CurrentWork.OperatingProject.ZoningWay += XY;
    Symbol.MultiSymbol_1 += ZoningScale.Text;
}
```

Rules for examining resources

```
ruleset CraneTypeRules {}

fcrule SuggestionPriorityRule in CraneTypeRules
{if:
    ?suggestion = CurrentWork.OperatingZone.Resource..UserSuggestion;
    IsObject(?suggestion);
    ?suggestion.ClassLevel == 2;
```

```

then:
    CurrentWork.OperatingZone.Resource..ResourceType
    = CurrentWork.OperatingZone.Resource..UserSuggestion;
    Symbol.MultiSymbol_1 += CTypeSPriority.Text1;
}

fcrule HeightRule in CraneTypeRules
{if:
    CurrentWork.OperatingZone.Resource..ExTimes == 1;
    CurrentWork.OperatingZone.Max_Z >= CNormal.HeightLimit;
then:
    CurrentWork.OperatingZone.Resource..ResourceType == TruckCranes;
    CurrentWork.OperatingZone.Resource..ResourceType == CrawlerCranes;
    Symbol.MultiSymbol_1 += CTypeHeight.Text1;
}

fcrule StoryRule in CraneTypeRules
{if:
    CurrentWork.OperatingZone.Resource..ExTimes == 1;
    CurrentWork.OperatingZone.PositionMax_Z >= CNormal.StoryLimit;
then:
    CurrentWork.OperatingZone.Resource..ResourceType == TruckCranes;
    CurrentWork.OperatingZone.Resource..ResourceType == CrawlerCranes;
    Symbol.MultiSymbol_1 += CTypeHeight.Text2;
}

fcrule RoomRule in CraneTypeRules
{if:
    CurrentWork.OperatingZone.Resource..ExTimes == 1;
    ?proj = CurrentWork.OperatingProject;
    ?proj.Site.XMax-?proj.Footprint.XMax < CNormal.RoomLimit;
    ?proj.Footprint.XMin-?proj.Site.XMin < CNormal.RoomLimit;
    ?proj.Site.YMax-?proj.Footprint.YMax < CNormal.RoomLimit;
    ?proj.Footprint.YMin-?proj.Site.YMin < CNormal.RoomLimit;
then:
    CurrentWork.OperatingZone.Resource..ResourceType == CrawlerCranes;
    Symbol.MultiSymbol_1 += CTypeEx.Text3;
}

fcrule PieceNoRule in CraneTypeRules
{if:
    CurrentWork.OperatingZone.Resource..ExTimes == 1;
    ?zone = CurrentWork.OperatingZone;
    ?no = ?zone.SBeamNo+?zone.SColumnNo+?zone.SRCBeamNo+?zone.SRCColumnNo;
    ?no > CNormal.PieceNo*1.5;
then:
    CurrentWork.OperatingZone.Resource..ResourceType == CrawlerCranes;
    CurrentWork.OperatingZone.Resource..ResourceType == TruckCranes;
    Symbol.MultiSymbol_1 += CTypeEx.Text4;
}

fcrule CraneUseRule in CraneTypeRules
{if:
    CurrentWork.OperatingZone.Resource..ExTimes == 1;
    ?zone = CurrentWork.OperatingZone;
    ?zone.OtherCraneUse == CNormal.CraneUse;
then:
    CurrentWork.OperatingZone.Resource..ResourceType == CrawlerCranes;
    CurrentWork.OperatingZone.Resource..ResourceType == TruckCranes;
    Symbol.MultiSymbol_1 += CTypeEx.Text5;
}

fcrule ComboRule1 in CraneTypeRules
{if:
    ?zone = CurrentWork.OperatingZone;
    ?zone.Resource..ExTimes == 1;
    ?zone.Max_Z < CNormal.HeightLimit-15.0;
    ?zone.Footprint.Area < CNormal.FpAreaLimit;
    ?zone.LongSide < CNormal.SideLimit;
    ?no = ?zone.SBeamNo+?zone.SColumnNo+?zone.SRCBeamNo+?zone.SRCColumnNo;
    ?no <= CNormal.PieceNo;
then:
    CurrentWork.OperatingZone.Resource..ResourceType == TowerCranes;
    Symbol.MultiSymbol_1 += CTypeEx.Text1;
}

fcrule ComboRule2 in CraneTypeRules
{if:
    CurrentWork.OperatingZone.Resource..ExTimes == 1;
    ?proj = CurrentWork.OperatingProject;
    ?proj.Site.XMax-?proj.Footprint.XMax > CNormal.RoomLimit+5.0;
    ?proj.Footprint.XMin-?proj.Site.XMin > CNormal.RoomLimit+5.0;
    ?proj.Site.YMax-?proj.Footprint.YMax > CNormal.RoomLimit+5.0;
    ?proj.Footprint.YMin-?proj.Site.YMin > CNormal.RoomLimit+5.0;
    CurrentWork.OperatingZone.Max_Z < CNormal.HeightLimit;
then:
    CurrentWork.OperatingZone.Resource..ResourceType == TowerCranes;
    CurrentWork.OperatingZone.Resource..ResourceType == TruckCranes;
    Symbol.MultiSymbol_1 += CTypeEx.Text2;
}

fcrule ConferingTaskSequenceRule1 in CraneTypeRules

```

```

{if:
    ?zone = CurrentWork.OperatingZone;
    ?zone.CriteriaImportanceR < ?zone.CriteriaImportanceT;
    ?zone.TaskSequence.OrderingPolicy == z;
    ListLength(all CurrentWork.OperatingZone.TaskSequence.OrderingPolicy)
        == 1;
then:
    ?zone.Resource..ResourceType -== TruckCranes;
    ?zone.Resource..ResourceType -== CrawlerCranes;
    Symbol.MultiSymbol_1 +== CTypeSeqPriority.Text1;
}

fcrule ConferingTaskSequenceRule2 in CraneTypeRules
{if:
    ?zone = CurrentWork.OperatingZone;
    ?zone.CriteriaImportanceR < ?zone.CriteriaImportanceT;
    for ?orderingpolicy = ?zone.TaskSequence.OrderingPolicy;
    always ?orderingpolicy != z;
then:
    ?zone.Resource..ResourceType -== TowerCranes;
    Symbol.MultiSymbol_1 +== CTypeSeqPriority.Text2;
}

fcrule TrCCCRule1 in CraneTypeRules
{if:
    ?zone = CurrentWork.OperatingZone;
    ?zone.Resource..ExTimes == 2;
    all CurrentWork.OperatingZone.Resource..ResourceType
        == `(CrawlerCranes,TruckCranes);
    ?zone.Site.Area < CNormal.SiteAreaLimit;
then:
    ?zone.Resource..ResourceType -== CrawlerCranes;
}

fcrule TrCCCRule2 in CraneTypeRules
{if:
    ?zone = CurrentWork.OperatingZone;
    ?zone.Resource..ExTimes == 2;
    all CurrentWork.OperatingZone.Resource..ResourceType
        == `(CrawlerCranes,TruckCranes);
    ?zone.Site.Area >= CNormal.SiteAreaLimit;
then:
    ?zone.Resource..ResourceType -== TruckCranes;
}

fcrule TCTrCRule1 in CraneTypeRules
{if:
    ?zone = CurrentWork.OperatingZone;
    ?zone.Resource..ExTimes == 2;
    all CurrentWork.OperatingZone.Resource..ResourceType
        == `(TruckCranes,TowerCranes);
    ?zone.ArchitecturalArea < CNormal.ArchAreaLimit;
then:
    ?zone.Resource..ResourceType -== TowerCranes;
}

fcrule TCTrCRule2 in CraneTypeRules
{if:
    ?zone = CurrentWork.OperatingZone;
    ?zone.Resource..ExTimes == 2;
    all CurrentWork.OperatingZone.Resource..ResourceType
        == `(TruckCranes,TowerCranes);
    ?zone.ArchitecturalArea >= CNormal.ArchAreaLimit;
then:
    ?zone.Resource..ResourceType -== TruckCranes;
}

fcrule TCTrCCCRule1 in CraneTypeRules
{if:
    ?zone = CurrentWork.OperatingZone;
    ?zone.Resource..ExTimes == 2;
    all CurrentWork.OperatingZone.Resource..ResourceType
        == `(CrawlerCranes,TruckCranes,TowerCranes);
    ?zone.ArchitecturalArea >= CNormal.ArchAreaLimit;
then:
    ?zone.Resource..ResourceType -== TruckCranes;
    ?zone.Resource..ResourceType -== CrawlerCranes;
}

fcrule TCTrCCCRule2 in CraneTypeRules
{if:
    ?zone = CurrentWork.OperatingZone;
    ?zone.Resource..ExTimes == 2;
    all CurrentWork.OperatingZone.Resource..ResourceType
        == `(CrawlerCranes,TruckCranes,TowerCranes);
    ?zone.ArchitecturalArea < CNormal.ArchAreaLimit;
    ?zone.Site.Area >= CNormal.SiteAreaLimit;
then:
    ?zone.Resource..ResourceType -== TruckCranes;
    ?zone.Resource..ResourceType -== TowerCranes;
}

```

```
fcrule TCTrCCCRule3 in CraneTypeRules
{if:
    ?zone = CurrentWork.OperatingZone;
    ?zone.Resource..ExTimes == 2;
    all CurrentWork.OperatingZone.Resource..ResourceType
        == (CrawlerCranes,TruckCranes,TowerCranes);
    ?zone.ArchitecturalArea < CNormal.ArchAreaLimit;
    ?zone.Site.Area < CNormal.SiteAreaLimit;
then:
    ?zone.Resource..ResourceType -== CrawlerCranes;
    ?zone.Resource..ResourceType -== TowerCranes;
}
```

Rules for examining task sequence

```
ruleset OrderingPolicyRules {}
ruleset OrderingPolicyHDetailRules {}
```

```
fcrule HeightRule in OrderingPolicyRules
{if:
    CurrentWork.OperatingZone.TaskSequence..ExTimes == 1;
    CurrentWork.OperatingZone.Max_Z >= SNormal.HeightLimit;
then:
    CurrentWork.OperatingZone.TaskSequence.HorizontalSequence = No;
    Symbol.MultiSymbol_1 +== OrdPHeight.Text1;
}
```

```
fcrule StoryRule in OrderingPolicyRules
{if:
    CurrentWork.OperatingZone.TaskSequence..ExTimes == 1;
    CurrentWork.OperatingZone.PositionMax_Z >= SNormal.StoryLimit;
then:
    CurrentWork.OperatingZone.TaskSequence.HorizontalSequence = No;
    Symbol.MultiSymbol_1 +== OrdPHeight.Text2;
}
```

```
fcrule FPAreaRule in OrderingPolicyRules
{if:
    CurrentWork.OperatingZone.TaskSequence..ExTimes == 1;
    CurrentWork.OperatingZone.Footprint.Area < SNormal.FootprintAreaLimit;
then:
    CurrentWork.OperatingZone.TaskSequence.VirtualSequence = No;
    Symbol.MultiSymbol_1 +== OrdPAreaSmall.Text1;
}
```

```
fcrule AAreaRule in OrderingPolicyRules
{if:
    CurrentWork.OperatingZone.TaskSequence..ExTimes == 1;
    CurrentWork.OperatingZone.ArchitecturalArea < SNormal.ArchitecturalAreaLimit;
then:
    CurrentWork.OperatingZone.TaskSequence.VirtualSequence = No;
    Symbol.MultiSymbol_1 +== OrdPAreaSmall.Text2;
}
```

```
fcrule IndependencyRule1 in OrderingPolicyRules
{if:
    CurrentWork.OperatingZone.TaskSequence..ExTimes == 1;
    CurrentWork.OperatingZone.SRCBeamNo+CurrentWork.OperatingZone.SRCColumnNo != 0;
    CurrentWork.OperatingZone.Max_Z/CurrentWork.OperatingZone.ShortSide >= SNormal.HDRatio;
    CurrentWork.OperatingZone.ShortSide ==
        CurrentWork.OperatingZone.Max_X-CurrentWork.OperatingZone.Min_X;
    CurrentWork.OperatingZone.PositionMax_X < 4.0;
then:
    CurrentWork.OperatingZone.TaskSequence.HorizontalSequence = No;
    Symbol.MultiSymbol_1 +== OrdPIndep.Text;
}
```

```
fcrule IndependencyRule2 in OrderingPolicyRules
{if:
    CurrentWork.OperatingZone.TaskSequence..ExTimes == 1;
    CurrentWork.OperatingZone.SRCBeamNo+CurrentWork.OperatingZone.SRCColumnNo != 0;
    CurrentWork.OperatingZone.Max_Z/CurrentWork.OperatingZone.ShortSide >= SNormal.HDRatio;
    CurrentWork.OperatingZone.ShortSide ==
        CurrentWork.OperatingZone.Max_Y-CurrentWork.OperatingZone.Min_Y;
    CurrentWork.OperatingZone.PositionMax_Y < 4.0;
then:
    CurrentWork.OperatingZone.TaskSequence.HorizontalSequence = No;
    Symbol.MultiSymbol_1 +== OrdPIndep.Text;
}
```

```
fcrule CraneUseRule in OrderingPolicyRules
{if:
    CurrentWork.OperatingZone.TaskSequence..ExTimes == 1;
    CurrentWork.OperatingZone.OtherCraneUse == SNormal.OtherCraneUse;
then:
    CurrentWork.OperatingZone.TaskSequence.HorizontalSequence
        = No; Symbol.MultiSymbol_1 +== OrdPCraneUse.Text;
}
```

```
fcrule ExcludeHRule in OrderingPolicyRules
{if:
```

```

    CurrentWork.OperatingZone.TaskSequence..ExTimes == 1;
    CurrentWork.OperatingZone.TaskSequence.HorizontalSequence == No;
then:
    CurrentWork.OperatingZone.TaskSequence.OrderingPolicy -== x;
    CurrentWork.OperatingZone.TaskSequence.OrderingPolicy -== y;
    CurrentWork.OperatingZone.TaskSequence.OrderingPolicy -== rx;
    CurrentWork.OperatingZone.TaskSequence.OrderingPolicy -== ry;
}

fcrule ExcludeVRule in OrderingPolicyRules
{if:
    CurrentWork.OperatingZone.TaskSequence..ExTimes == 1;
    CurrentWork.OperatingZone.TaskSequence.VirtualSequence == No;
then:
    CurrentWork.OperatingZone.TaskSequence.OrderingPolicy -== z;
}

fcrule ExcludeYRule in OrderingPolicyHDetailRules
{if:
    CurrentWork.OperatingZone.TaskSequence..ExTimes == 1;
    CurrentWork.OperatingZone.TaskSequence.HorizontalSequence == Yes;
    CurrentWork.OperatingZone.LongSide ==
        CurrentWork.OperatingZone.Max_X-CurrentWork.OperatingZone.Min_X;
    CurrentWork.OperatingZone.LongSide/CurrentWork.OperatingZone.ShortSide >= 1.5;
then:
    CurrentWork.OperatingZone.TaskSequence.OrderingPolicy -== y;
    CurrentWork.OperatingZone.TaskSequence.OrderingPolicy -== ry;
    Symbol.MultiSymbol_1 +== OrdPLSRatio.Text1;
}

fcrule ExcludeXRule in OrderingPolicyHDetailRules
{if:
    CurrentWork.OperatingZone.TaskSequence..ExTimes == 1;
    CurrentWork.OperatingZone.TaskSequence.HorizontalSequence == Yes;
    CurrentWork.OperatingZone.LongSide ==
        CurrentWork.OperatingZone.Max_Y-CurrentWork.OperatingZone.Min_Y;
    CurrentWork.OperatingZone.LongSide/CurrentWork.OperatingZone.ShortSide >= 1.5;
then:
    CurrentWork.OperatingZone.TaskSequence.OrderingPolicy -== x;
    CurrentWork.OperatingZone.TaskSequence.OrderingPolicy -== rx;
    Symbol.MultiSymbol_1 +== OrdPLSRatio.Text2;
}

fcrule XRRule in OrderingPolicyHDetailRules
{if:
    CurrentWork.OperatingZone.TaskSequence..ExTimes == 1;
    CurrentWork.OperatingZone.TaskSequence.HorizontalSequence == Yes;
    CurrentWork.OperatingZone.GateName.Location..X
        >= (CurrentWork.OperatingZone.Max_X-CurrentWork.OperatingZone.Min_X)/2.0
        +CurrentWork.OperatingZone.Min_X;
then:
    CurrentWork.OperatingZone.TaskSequence.OrderingPolicy -== rx;
    Symbol.MultiSymbol_1 +== OrdPGLocation.Text1;
}

fcrule RXRule in OrderingPolicyHDetailRules
{if:
    CurrentWork.OperatingZone.TaskSequence..ExTimes == 1;
    CurrentWork.OperatingZone.TaskSequence.HorizontalSequence == Yes;
    CurrentWork.OperatingZone.GateName.Location..X
        < (CurrentWork.OperatingZone.Max_X-CurrentWork.OperatingZone.Min_X)/2.0
        +CurrentWork.OperatingZone.Min_X;
then:
    CurrentWork.OperatingZone.TaskSequence.OrderingPolicy -== x;
    Symbol.MultiSymbol_1 +== OrdPGLocation.Text2;
}

fcrule YRule in OrderingPolicyHDetailRules
{if:
    CurrentWork.OperatingZone.TaskSequence..ExTimes == 1;
    CurrentWork.OperatingZone.TaskSequence.HorizontalSequence == Yes;
    CurrentWork.OperatingZone.GateName.Location..Y
        >= (CurrentWork.OperatingZone.Max_Y-CurrentWork.OperatingZone.Min_Y)/2.0
        +CurrentWork.OperatingZone.Min_Y;
then:
    CurrentWork.OperatingZone.TaskSequence.OrderingPolicy -== ry;
    Symbol.MultiSymbol_1 +== OrdPGLocation.Text3;
}

fcrule RYRule in OrderingPolicyHDetailRules
{if:
    CurrentWork.OperatingZone.TaskSequence..ExTimes == 1;
    CurrentWork.OperatingZone.TaskSequence.HorizontalSequence == Yes;
    CurrentWork.OperatingZone.GateName.Location..Y
        < (CurrentWork.OperatingZone.Max_Y-CurrentWork.OperatingZone.Min_Y)/2.0
        +CurrentWork.OperatingZone.Min_Y;
then:
    CurrentWork.OperatingZone.TaskSequence.OrderingPolicy -== y;
    Symbol.MultiSymbol_1 +== OrdPGLocation.Text4;
}

fcrule ResourceConferringRule1 in OrderingPolicyRules

```

COMPAS: Application of Knowledge-Based System to Construction Planning

```
{if:
    ?zone = CurrentWork.OperatingZone;
    ?zone.CriteriaImportanceR >= ?zone.CriteriaImportanceT;
    ?zone.Resource..ResourceType == TowerCranes;
    ListLength(all CurrentWork.OperatingZone.Resource) == 1;
then:
    ?zone.TaskSequence.OrderingPolicy == x;
    ?zone.TaskSequence.OrderingPolicy == rx;
    ?zone.TaskSequence.OrderingPolicy == y;
    ?zone.TaskSequence.OrderingPolicy == ry;
    Symbol.MultiSymbol_1 += OrdPRPriority.Text1;
}

fcrule ResourceConferringRule2 in OrderingPolicyRules
{if:
    ?zone = CurrentWork.OperatingZone;
    ?zone.CriteriaImportanceR >= ?zone.CriteriaImportanceT;
    for ?resource = ?zone.Resource..ResourceType;
    always ?resource != TowerCranes;
then:
    ?zone.TaskSequence.OrderingPolicy == z;
    Symbol.MultiSymbol_1 += OrdPRPriority.Text2;
}

fcrule FinalRule in OrderingPolicyRules
{if:
    ?zone = CurrentWork.OperatingZone;
    ?zone.TaskSequence..ExTimes == 2;
then:
    ?zone.TaskSequence.OrderingPolicy == z;
    ?zone.TaskSequence.OrderingPolicy == y;
    ?zone.TaskSequence.OrderingPolicy == ry;
}

ruleset ErectionSequenceRules {}
ruleset StatusNotReadyRules {ErectionSequenceRules}
ruleset ResettoOkRules {ErectionSequenceRules}
ruleset DefaultBestRules {ErectionSequenceRules}
ruleset ResetRules {ErectionSequenceRules}

fcrule ActivityDependencyRule in StatusNotReadyRules
{if:
    ?activity = CurrentWork.OperatingZone.Activity;
    not ?activity.Sequencing_Status == sequenced;
    ?pred = ?activity.Preceding_Activity;
    not ?pred.Sequencing_Status == sequenced;
then:
    ?activity.Sequencing_Status = not_ready;
}

fcrule ResettoOkRule in ResettoOkRules
{if:
    ?activity = CurrentWork.OperatingZone.Activity;
    not ?activity.Sequencing_Status == sequenced;
then:
    ?activity.Sequencing_Status = okay;
}

fcrule StabilityRule in StatusNotReadyRules
{if:
    ?activity = CurrentWork.OperatingZone.Activity;
    ?activity.Sequencing_Status != sequenced;
    ?obj1 = ?activity.Involved_Object;
    classof ?obj1 == Columns;
    ?obj2 = ?obj1.Supported_by;
    classof ?obj2 == Columns;
    ?activity2 = ?obj2.Activity_of;
    ?activity2.Sequencing_Status == sequenced;
    not find{?obj3.Supported_by == ?obj2;
        classof ?obj3 == Beams;
        ?obj4.Supported_by == ?obj2;
        classof ?obj4 == Beams;
        ?obj3 != ?obj4;
        ?activity3 = ?obj3.Activity_of;
        ?activity4 = ?obj4.Activity_of;
        ?activity3.Sequencing_Status == sequenced;
        ?activity4.Sequencing_Status == sequenced;
        ?obj3.Position..Begin_X == ?obj3.Position..End_X;
        ?obj4.Position..Begin_Y == ?obj4.Position..End_Y;
    }
then:
    ?activity.Sequencing_Status = not_ready;
}

fcrule DefaultBestRule in DefaultBestRules
{if:
    ?activity = CurrentWork.OperatingZone.Activity;
    ?activity.Sequencing_Status == okay;
    not find {?act = CurrentWork.OperatingZone.Activity;
        ?act.Sequencing_Status == best;}
then:
```



```

    ?activity.Sequencing_Status = best;
}

```

Rules for examining lay out

```
ruleset ExamineLayoutRules {}
```

```
fcrule SuggestionPriorityRule in ExamineLayoutRules
```

```
{if:
    ?zone = CurrentWork.OperatingZone;
    ?layout = ?zone.Layout;
    IsSymbol(?zone.Layout..UserSuggestion);
    ?zone.Layout..UserSuggestion != none;
then:
    ?layout.Locations += ?zone.Layout..UserSuggestion;
    ?layout.ExaminingStatus = examined;
    Symbol.MultiSymbol_1 += ExLOSPriority.Text1;
}
```

```
fcrule ExamineOutsideRule in ExamineLayoutRules
```

```
{if:
    ?zone = CurrentWork.OperatingZone;
    ?lay = ?zone.Layout; ?zone.Layout..UserSuggestion == none;
    ?lay.Location == Outside; ?lay.ExaminingStatus != examined;
    ?lay.ExaminingStatus != being examined;
    ?lay.ExaminingStatus = being examined;
    ?grid = find instanceof GRIDS;
    ?grid.Occupant == find instanceof Sites;
    ?neighbor = SendMsg(?grid,FindNeighbors!,?zone.Footprint);
    ListLength(?neighbor) >= 1;
then:
    ?lay.Locations += ?grid;
    ?lay.Locations..LocationsProcess += ?grid;
    ?lay.ExaminingStatus = being_examined;
}
```

```
fcrule ExamineInsideRule in ExamineLayoutRules
```

```
{if:
    ?zone = CurrentWork.OperatingZone;
    ?zone.Layout..UserSuggestion == none;
    ?zone.Layout.Location == Outside;
    ?zone.Layout.ExaminingStatus == examined;
    ?zone.Layout.Locations == `();
then:
    ?layoutlist = all ?zone.Layout;
    for ?layout inlist ?layoutlist;
        do {?layout.Location = Inside;
            ?layout.ExaminingStatus = Null;}
}
```

```
fcrule TCOutsideRule in ExamineLayoutRules
```

```
{if:
    ?zone = CurrentWork.OperatingZone;
    ?lay = ?zone.Layout;
    ?zone.Layout..UserSuggestion == none;
    ?zone.Resource == direct instanceof TowerCranes;
    ?lay.Location == Outside;
    ?lay.ExaminingStatus == being examined;
    ?layoutlist = all CurrentWork.OperatingZone.Layout.Locations;
    ?lay.ExaminingStatus = examined;
    find ?layout inlist ?layoutlist;
    KittyCorner(?zone.GateName.Location..X,?zone.GateName.Location..Y,
        (?zone.Max_X-?zone.Min_X)/2.0+?zone.Min_X,
        (?zone.Max_Y-?zone.Min_Y)/2.0+?zone.Min_Y,
        ?layout.Center_X,?layout.Center_Y) == -1;
then:
    ?lay.Locations -= ?layout;
    ?lay.Locations..LocationsProcess -= ?layout;
    ?text = ConvertToSymbol
        (AppendStrings("Text",ConvertToString(Numeric.Numeric_5)));
    Symbol.MultiSymbol_1 += ExLOTOut.?text;
    ?lay.ExaminingStatus = examined;
}
```

```
fcrule TCInsideRule in ExamineLayoutRules
```

```
{if:
    ?zone = CurrentWork.OperatingZone;
    ?zone.Layout..UserSuggestion == none;
    ?zone.Resource == direct instanceof TowerCranes;
    ?zone.Layout.Location == Inside;
then:
    ?x = (?zone.Max_X-?zone.Min_X)/2.0+?zone.Min_X;
    ?y = (?zone.Max_Y-?zone.Min_Y)/2.0+?zone.Min_Y;
    for find ?grid = instanceof GRIDS;
        do {?nearestgrid = SendMsg(?grid,FindNearestGrid!,?x,?y);
            ?nearestgrid != Null; ?ngrid = ?nearestgrid;}
    ?layoutlist2 = all ?zone.Layout;
    for ?layout2 inlist ?layoutlist2;
        do {?layout2.Locations += ?ngrid;
            ?layout2.Locations..LocationsProcess += ?ngrid;}
    Symbol.MultiSymbol_1 += ExLOTIn.Text1;
}
```

```

}
fcrule MovableLocationRule in ExamineLayoutRules
{if:
    ?zone = CurrentWork.OperatingZone;
    ?zone.Layout..UserSuggestion == none;
    or {?zone.Resource == direct instanceof TruckCranes;
        ?zone.Resource == direct instanceof CrawlerCranes;}
    ?zone.Layout.Location == Inside;
then:
    ?layoutlist2 = all ?zone.Layout;
    for ?layout2 inlist ?layoutlist2;
    do ?layout2.Location..Movable = Yes;
}

fcrule TrCCCOutsideRule in ExamineLayoutRules
{if:
    ?zone = CurrentWork.OperatingZone;
    ?lay = ?zone.Layout;
    ?lay.ExaminingStatus == being_examined;
    ?zone.Layout..UserSuggestion == none;
    or {?zone.Resource == direct instanceof TruckCranes;
        ?zone.Resource == direct instanceof CrawlerCranes;}
    ?lay.Location == Outside; ?lay.ExaminingStatus = examined;
    ?layoutlist = all CurrentWork.OperatingZone.Layout.Locations;
    find ?layout inlist ?layoutlist;
    AdjacencyPreference(?zone.TaskSequence.OrderingPolicy,
        ?layout.Center X, ?layout.Center Y,
        (?zone.Max_X-?zone.Min_X)/2+?zone.Min_X,
        (?zone.Max_Y-?zone.Min_Y)/2+?zone.Min_Y) == -1;
then:
    ?lay.Locations -== ?layout;
    ?lay.Locations..LocationsProcess -== ?layout;
    ?text = ConvertToSymbol
        (AppendStrings("Text", ConvertToString(Numeric.Numeric_5)));
    Symbol.MultiSymbol 1 +== ExLOTrCCCOut.?text;
    ?lay.ExaminingStatus = examined;
}

fcrule FPAreaLimitRule in ExamineLayoutRules
{if:
    ?zone = CurrentWork.OperatingZone;
    ?zone.Layout..UserSuggestion == none;
    ?zone.Layout.ExaminingStatus == not_examined;
    ?resource = ?zone.Layout.Resource;
    ?resourcetype = find direct classof ?resource;
then:
    ?inout = JudgeInOut(?resource);
    ?layoutlist2 = all ?zone.Layout;
    for ?layout2 inlist ?layoutlist2;
    do ?layout2.Location = ?inout;
    ?text = ConvertToSymbol
        (AppendStrings("Text", ConvertToString(Numeric.Numeric_5)));
    Symbol.MultiSymbol 1 +==
        AppendStrings(ExLOInOut.?text, ConvertToString(?resourcetype));
}

fcrule MovableRule in ExamineLayoutRules
{if:
    ?zone = CurrentWork.OperatingZone;
    ?layout = ?zone.Layout;
    ?zone.Layout..UserSuggestion == none;
    ?layout.Location..Movable == Yes;
    ?layout.Location == Inside;
    ?layout.ExaminingStatus != examined;
    find ?resource = ?zone.Resource;
    find ?resourcetype = direct classof ?resource;
    ?number = NoofZones(?resourcetype, ?zone.TaskSequence.OrderingPolicy);
    ?number != Null;
then:
    CreateMultiLayout(?number-1, Inside, Yes);
    ?sequence = ?zone.TaskSequence.OrderingPolicy;
    AssignInsideLocation(?sequence, ?resourcetype, ?zone, ?number);
}

fcrule ClimbingRule in ExamineLayoutRules
{if:
    ?zone = CurrentWork.OperatingZone;
    find ?resource = ?zone.Resource;
    ?resource == direct instanceof TowerCranes;
    ?inout = ?zone.Layout.Location;
then:
    ?crimblast = ClimbingTimes(?resource, ?zone.Max_Z, ?zone.ZCoordUnit);
    ?number = ListLength(?crimblast)/2.0;
    CreateMultiLayout(?number-1, ?inout, Null);
}

```

Rules for examining priority

```
ruleset CImportanceRules {}
```

```

fcrule SiteAreaRule in CImportanceRules
{if:
    ?zone = CurrentWork.OperatingZone;
    ?zone.Site.Area > CrNormal.SiteAreaUpper;
then:
    ?zone.CriteriaImportanceR = ?zone.CriteriaImportanceR+3;
    ?zone.CriteriaImportanceT = ?zone.CriteriaImportanceT+3;
}

fcrule SiteArea2Rule in CImportanceRules
{if:
    ?zone = CurrentWork.OperatingZone;
    ?zone.Site.Area < CrNormal.SiteAreaLower;
then:
    ?zone.CriteriaImportanceR = ?zone.CriteriaImportanceR+4;
    Symbol.MultiSymbol_1 += CImpAreaLower.Text1;
}

fcrule SiteRatioRule in CImportanceRules
{if:
    ?zone = CurrentWork.OperatingZone;
    ?zone.Site.Area > 2000.0;
    ?x = ?zone.Site.XMax-?zone.Site.XMin;
    ?y = ?zone.Site.YMax-?zone.Site.YMin;
    or {?x/?y > CrNormal.SiteLSRatio;
        ?y/?x > CrNormal.SiteLSRatio;}
then:
    ?zone.CriteriaImportanceT = ?zone.CriteriaImportanceT+2;
    Symbol.MultiSymbol_1 += CImpSiteRatio.Text1;
}

fcrule GateLocationRuleX in CImportanceRules
{if:
    ?zone = CurrentWork.OperatingZone;
    ?gatelist == all CurrentWork.OperatingProject.GateName;
    ListLength(?gatelist) == 1;
    ?gate = ListFirst(?gatelist);
    ?x = ?zone.Site.XMax-?zone.Site.XMin;
    ?y = ?zone.Site.YMax-?zone.Site.YMin;
    ?x >= ?y*1.5;
    ?l = ?x*CrNormal.GLocationRatio;
    or {?gate.Location..X < ?zone.Site.XMin+?l;
        ?gate.Location..X > ?zone.Site.XMax-?l;}
then:
    ?zone.CriteriaImportanceR = ?zone.CriteriaImportanceR+0;
    ?zone.CriteriaImportanceT = ?zone.CriteriaImportanceT+7;
    Symbol.MultiSymbol_1 += CImpGate.Text1;
}

fcrule GateLocationRuleY in CImportanceRules
{if:
    ?zone = CurrentWork.OperatingZone;
    ?gatelist == all CurrentWork.OperatingProject.GateName;
    ListLength(?gatelist) == 1;
    ?gate = ListFirst(?gatelist);
    ?x = ?zone.Site.XMax-?zone.Site.XMin;
    ?y = ?zone.Site.YMax-?zone.Site.YMin;
    ?y >= ?x*1.5;
    ?l = ?y*CrNormal.GLocationRatio;
    or {?gate.Location..Y < ?zone.Site.YMin+?l;
        ?gate.Location..Y > ?zone.Site.YMax-?l;}
then:
    ?zone.CriteriaImportanceR = ?zone.CriteriaImportanceR+0;
    ?zone.CriteriaImportanceT = ?zone.CriteriaImportanceT+7;
    Symbol.MultiSymbol_1 += CImpGate.Text1;
}

fcrule GateWidthRule in CImportanceRules
{if:
    ?zone = CurrentWork.OperatingZone;
    ?gate = ?zone.GateName; ?gate.Priority == 1;
    ?gate.Width <= CrNormal.GateWidth;
then:
    ?zone.CriteriaImportanceR = ?zone.CriteriaImportanceR+2;
    ?zone.CriteriaImportanceT = ?zone.CriteriaImportanceT+0;
    Symbol.MultiSymbol_1 += CImpGate.Text2;
}

fcrule FootprintCoverage1Rule in CImportanceRules
{if:
    ?zone = CurrentWork.OperatingZone;
    ?coverage = CurrentWork.OperatingProject.Footprint.Area/?zone.Site.Area;
    ?coverage < CrNormal.FPCoverageLower;
then:
    ?zone.CriteriaImportanceR = ?zone.CriteriaImportanceR+3;
    ?zone.CriteriaImportanceT = ?zone.CriteriaImportanceT+0;
    Symbol.MultiSymbol_1 += CImpFPC.Text1;
}

fcrule FootprintCoverage2Rule in CImportanceRules
{if:
    ?zone = CurrentWork.OperatingZone;

```

```

        ?coverage = CurrentWork.OperatingProject.Footprint.Area/?zone.Site.Area;
        ?coverage >= CrNormal.FPCoverageLower;
        ?coverage < CrNormal.FPCoverageUpper;
    then:
        ?zone.CriteriaImportanceR = ?zone.CriteriaImportanceR+1;
        Symbol.MultiSymbol_1 += CImpFPC.Text2;
    }
fcrule FootprintCoverage3Rule in CImportanceRules
{if:
    ?zone = CurrentWork.OperatingZone;
    ?coverage = CurrentWork.OperatingProject.Footprint.Area/?zone.Site.Area;
    ?coverage >= CrNormal.FPCoverageUpper;
then:
    ?zone.CriteriaImportanceR = ?zone.CriteriaImportanceR+2;
    Symbol.MultiSymbol_1 += CImpFPC.Text3;
}
fcrule FootprintSideRule in CImportanceRules
{if:
    ?zone = CurrentWork.OperatingZone;
    ?zone.ShortSide < CrNormal.FPSide;
then:
    ?zone.CriteriaImportanceT = ?zone.CriteriaImportanceT+2;
    Symbol.MultiSymbol_1 += CImpFPArea.Text6;
}
fcrule FootprintArealRule in CImportanceRules
{if:
    ?zone = CurrentWork.OperatingZone;
    ?zone.Footprint.Area > CrNormal.FPAreaUpper;
then:
    ?zone.CriteriaImportanceR = ?zone.CriteriaImportanceR+2;
    Symbol.MultiSymbol_1 += CImpFPArea.Text1;
}
fcrule FootprintArea2Rule in CImportanceRules
{if:
    ?zone = CurrentWork.OperatingZone;
    ?zone.Footprint.Area < CrNormal.FPAreaLower;
then:
    ?zone.CriteriaImportanceR = ?zone.CriteriaImportanceR+5;
    Symbol.MultiSymbol_1 += CImpFPArea.Text2;
}
fcrule FootprintRatioRule in CImportanceRules
{if:
    ?zone = CurrentWork.OperatingZone;
    ?zone.LongSide/?zone.ShortSide > CrNormal.FPLSRatio;
then:
    ?zone.CriteriaImportanceT = ?zone.CriteriaImportanceT+2;
    Symbol.MultiSymbol_1 += CImpFPRatio.Text1;
}
fcrule HeightRule in CImportanceRules
{if:
    ?zone = CurrentWork.OperatingZone;
    local (?piecelist = {});
    for find ?piece = ?zone.Component;
    do {?piece.TopColumn != 0;
        ?piece.Location..Z+?piece.Height > CrNormal.HeightLimit;
        collect ?piece into ?piecelist;}}
    ListLength(?piecelist) >= 1;
then:
    ?zone.CriteriaImportanceR = ?zone.CriteriaImportanceR+2;
    Symbol.MultiSymbol_1 += CImpHeight.Text1;
}
fcrule PieceNo1Rule in CImportanceRules
{if:
    ?zone = CurrentWork.OperatingZone;
    ?pieceno = ?zone.SColumnNo+?zone.SBeamNo+?zone.SRCColumnNo+?zone.SRCBeamNo;
    ?pieceno < CrNormal.PieceNoLower;
then:
    ?zone.CriteriaImportanceT = ?zone.CriteriaImportanceT+2;
    Symbol.MultiSymbol_1 += CImpPiece.Text1;
}
fcrule PieceNo2Rule in CImportanceRules
{if:
    ?zone = CurrentWork.OperatingZone;
    ?pieceno = ?zone.SColumnNo+?zone.SBeamNo+?zone.SRCColumnNo+?zone.SRCBeamNo;
    ?pieceno > CrNormal.PieceNoUpper;
then:
    ?zone.CriteriaImportanceR = ?zone.CriteriaImportanceR+2;
    Symbol.MultiSymbol_1 += CImpPiece.Text2;
}
fcrule PCRule in CImportanceRules
{if:
    ?zone = CurrentWork.OperatingZone;
    ?uselist = all ?zone.OtherCraneUse;

```