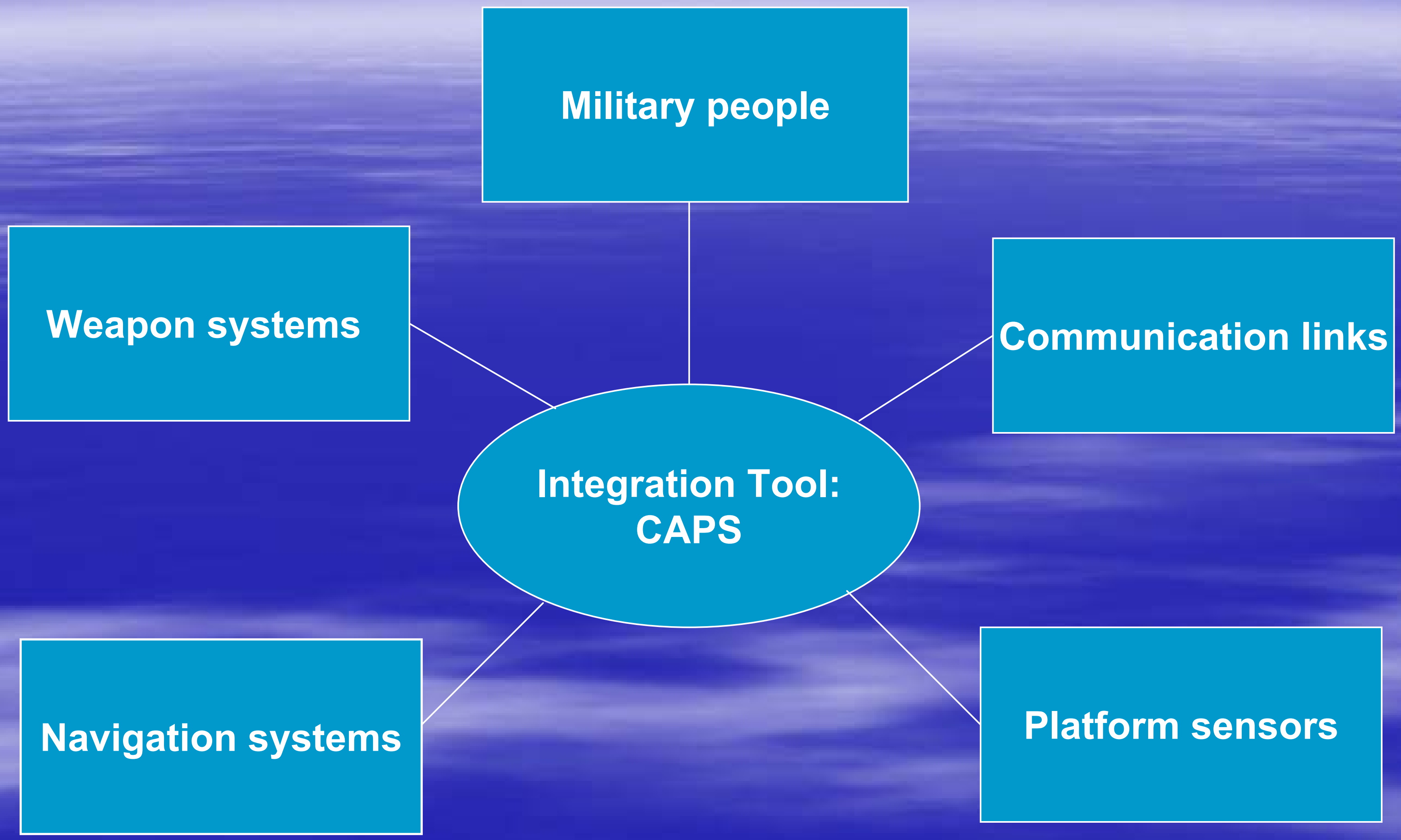
Integrating Battlefield Objects of C4ISR Systems by Using CAPS

Meng-chyi Harn
Cheng-hang Wang

Integrating Battlefield Objects of C4ISR Systems by Using CAPS

Dr. Meng-chyi Harn
Department of Information Management
China Institute of Technology
Taipei, Taiwan, R.O.C.

Col. Cheng-hang Wang
Department of Computer Science and Engineering
Yuan Ze University
Taoyuan, Taiwan, R.O.C.


Military people
Weapon systems
Communication links
Integration Tool:
CAPS
Navigation systems
Platform sensors

Requirements Description

- Changed by
  - National strategic thinking
  - Commander's operational needs
  - Battlefield resources allocations
- Hard to firm the user's requirements and specifications

Open Issues

- Capacity of a project leader and a system engineer
- Complexity of battlefield objects
- Formalization of military operation model
- Selection of a reliable and adaptive tool
- Additional issues in the system integration
  - Weapon system replacement
  - Real-time data communication
  - Business process reengineering
  - Command and control process changing

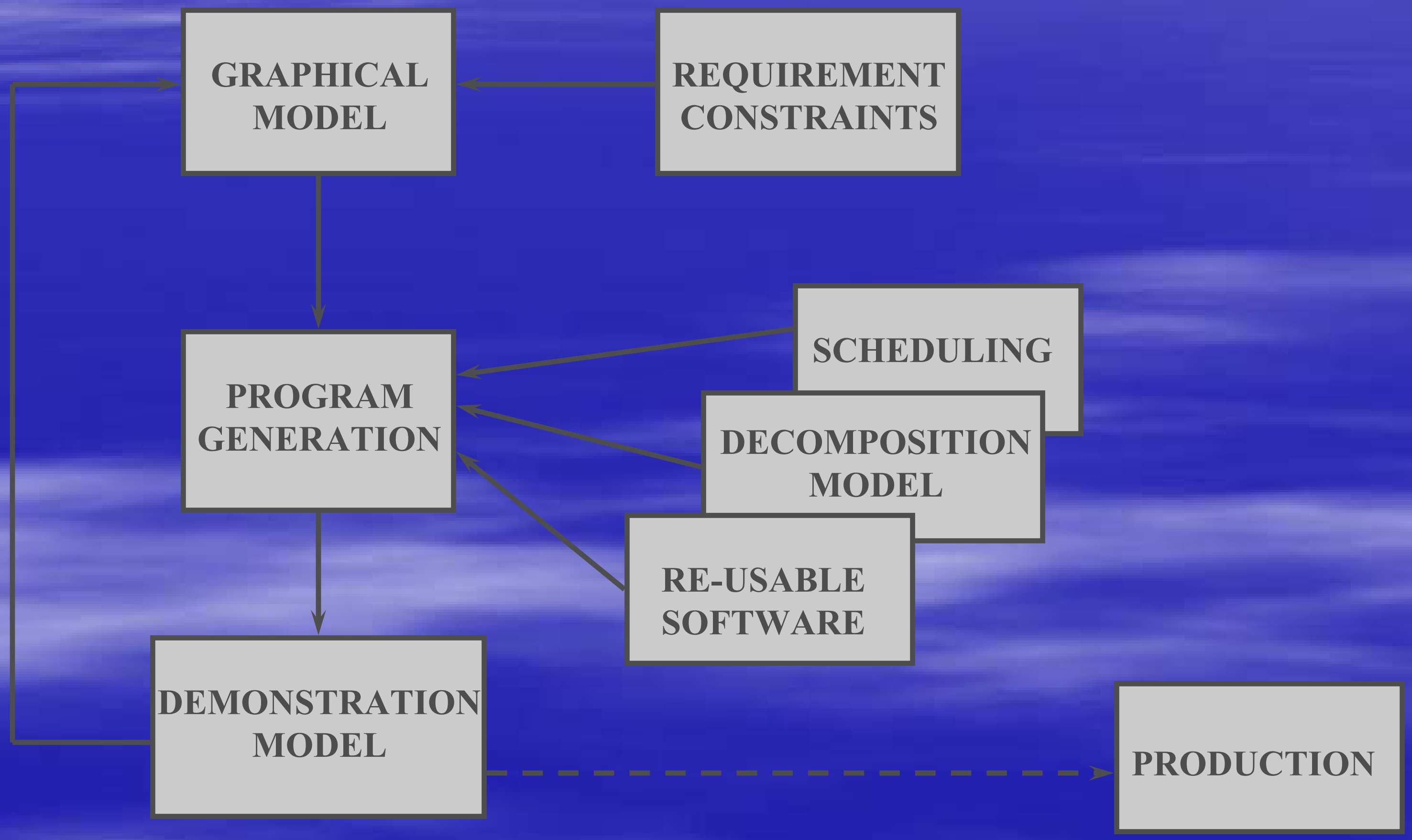
Our Solutions

- The rapid prototyping method
  - requirements analysis
  - feasibility studies
  - systematic development
- The formal model
  - CAPS

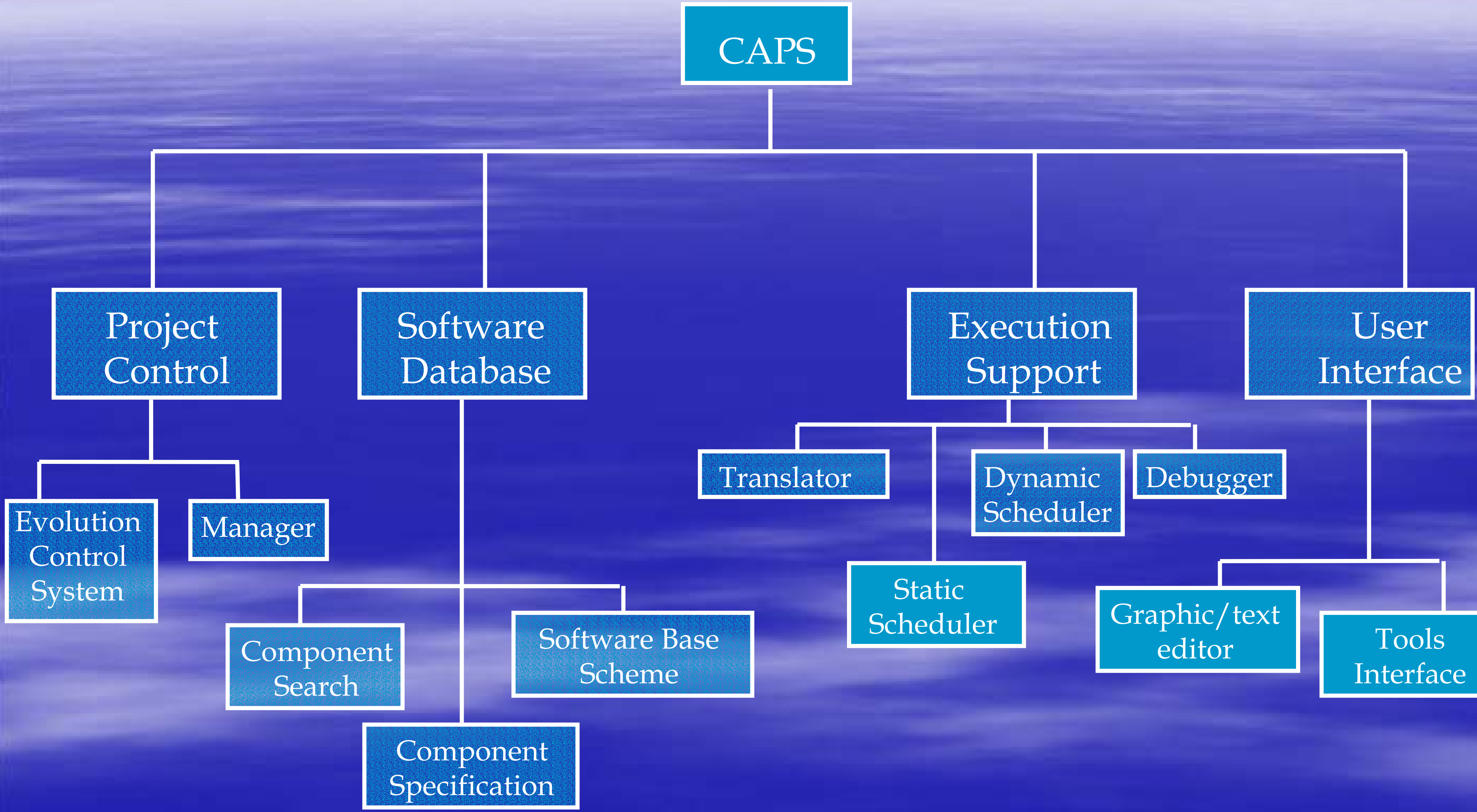
Why CAPS

- Rapidly evaluating requirements for real-time control software using executable prototypes
- Testing and integrating completed subsystems through evolutionary prototyping
- Quickly developing functional prototypes by a code generator

How CAPS Works



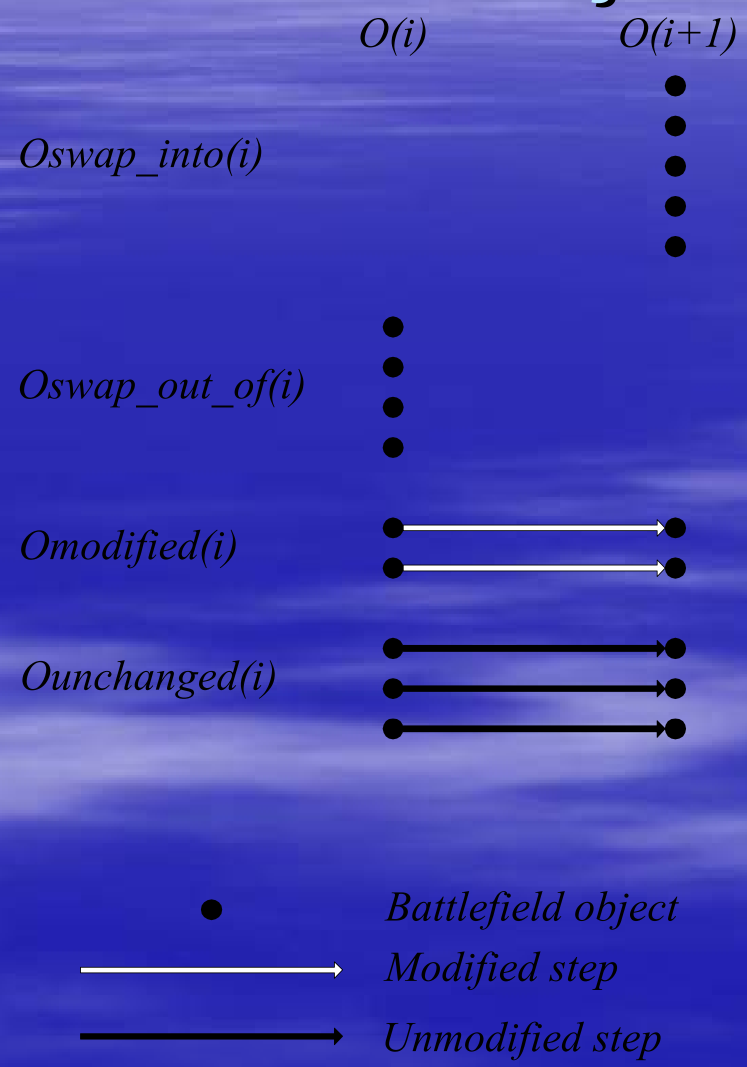
CAPS Architecture



Battlefield Objects

- $O = P \cup W \cup N \cup S \cup C$
  - P denotes a set of *military person objects*
  - W denotes a set of *weapon system objects*
  - N denotes a set of *navigation system objects*
  - S denotes a set of *platform sensor objects*
  - C denotes a set of *communication link objects*.

Integration Mechanism of Battlefield Objects



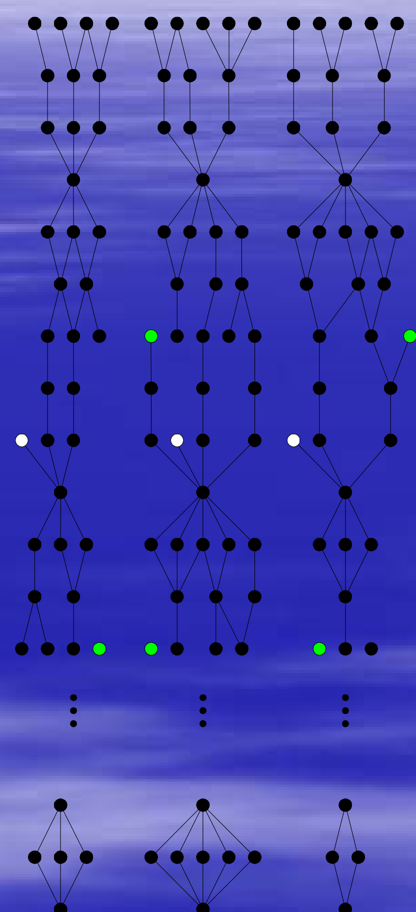
Evolution of Battlefield Objects

- $O(i+1) = O(i) + \Delta O(i)$
- $O(i+1) = P(i+1) \cup W(i+1) \cup N(i+1) \cup S(i+1) \cup C(i+1)$
- $O(i) = P(i) \cup W(i) \cup N(i) \cup S(i) \cup C(i)$
- $\Delta O(i) = \Delta P(i) \cup \Delta W(i) \cup \Delta N(i) \cup \Delta S(i) \cup \Delta C(i)$ where
  - $\Delta O(i) = Oswap\_into(i) – Oswap\_out\_of(i) \cup Omodified(i)$
  - $\Delta P(i) = Pswap\_into(i) – Pswap\_out\_of(i) \cup Pmodified(i)$
  - $\Delta W(i) = Wswap\_into(i) – Wswap\_out\_of(i) \cup Wmodified(i)$
  - $\Delta N(i) = Nswap\_into(i) – Nswap\_out\_of(i) \cup Nmodified(i)$
  - $\Delta S(i) = Sswap\_into(i) – Sswap\_out\_of(i) \cup Smodified(i)$
  - $\Delta C(i) = Cswap\_into(i) – Cswap\_out\_of(i) \cup Cmodified(i)$
- $P(i+1) = P(i) + \Delta P(i)$
- $W(i+1) = W(i) + \Delta W(i)$
- $N(i+1) = N(i) + \Delta N(i)$
- $S(i+1) = S(i) + \Delta S(i)$
- $C(i+1) = C(i) + \Delta C(i)$

Evolution of Battlefield Objects

Components

Requirements
Specifications
Modules
Porgrams
Criticisms
Issues
Requirements
Specifications
Modules
Porgrams
Criticisms
Issues
Requirements
⋮
Programs
Optimizations
Programs



Software Evolution Steps

Specification Design
Module Design
Software Prototype Integration
Software Prototype Demo
Issue Analysis
Requirements Analysis
Specification Design
Module Design
Software Prototype Integration
Software Prototype Demo
Issue Analysis
Requirements Analysis
⋮
Software Product Demo
Software Product Implementation

Versions

Prototype 1
Prototype 2
Prototype 3
⋮
Prototype n
Production Program 1

- ● *Software evolution component*
- ● *New user requirements component*
- ○ *Legacy program that is a set of module components*
- —— *Software evolution step*

Project Organization and Preparation

- The C4ISR system selected by our project is a *huge-grain* program [Luqi, 1997] involving module composition.
- We spent almost one year to conduct this project by nine project teams.
- Each project team had one project leader and 24 project members in charge of 25 *large-grain* programs that included one integrated meta-program and 24 individual programs.

Preliminary Courses

1. Digitizing Battlefield Management
   - Network Centric Warfare [Alberts *et al.*, 1999]
   - Understanding Information Age Warfare [Alberts *et al.*, 2001]
   - Power to the Edge [Alberts *et al.*, 2003]
   - Information Warfare and Security [Denning, 1999]
2. Development and Implementation of the C4ISR Systems
   - Command and Control Theory
   - Real-time Embedded Systems
   - Requirement Engineering and Rapid Prototyping
   - Computer-Aided Prototyping System
3. Evolution of the C4ISR Systems
   - Relational Hypergraph Model [Harn, 1999a]
   - Version Control and Configuration Management
   - Automated Software Engineering
   - Computer-Aided Software Engineering System [Harn, 1999a]

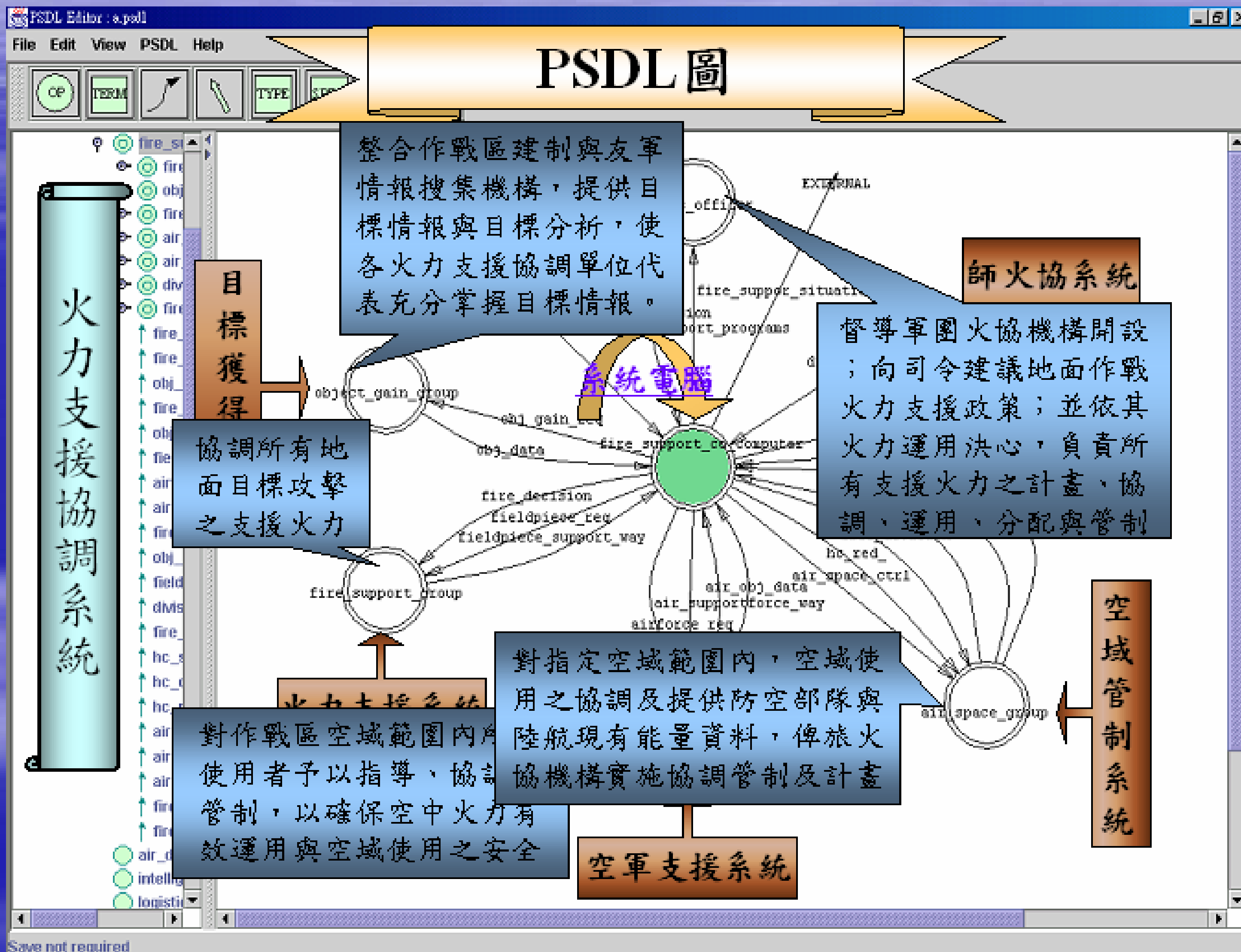
Operational Architecture

- Joint Operational Command and Control System (JOCCS)
- Decision Support System of Operation Area Commander (DSSOAC)
- Battle Command System (BCS)
- Intelligence Surveillance and Reconnaissance System (ISRS)
- Air Defense System (ADS)
- Fire Support and Coordination System (FSCS)
- Disaster Control System (DCS)
- Operational Service System (OSS)
- Personnel Information Integration System (PIIS)

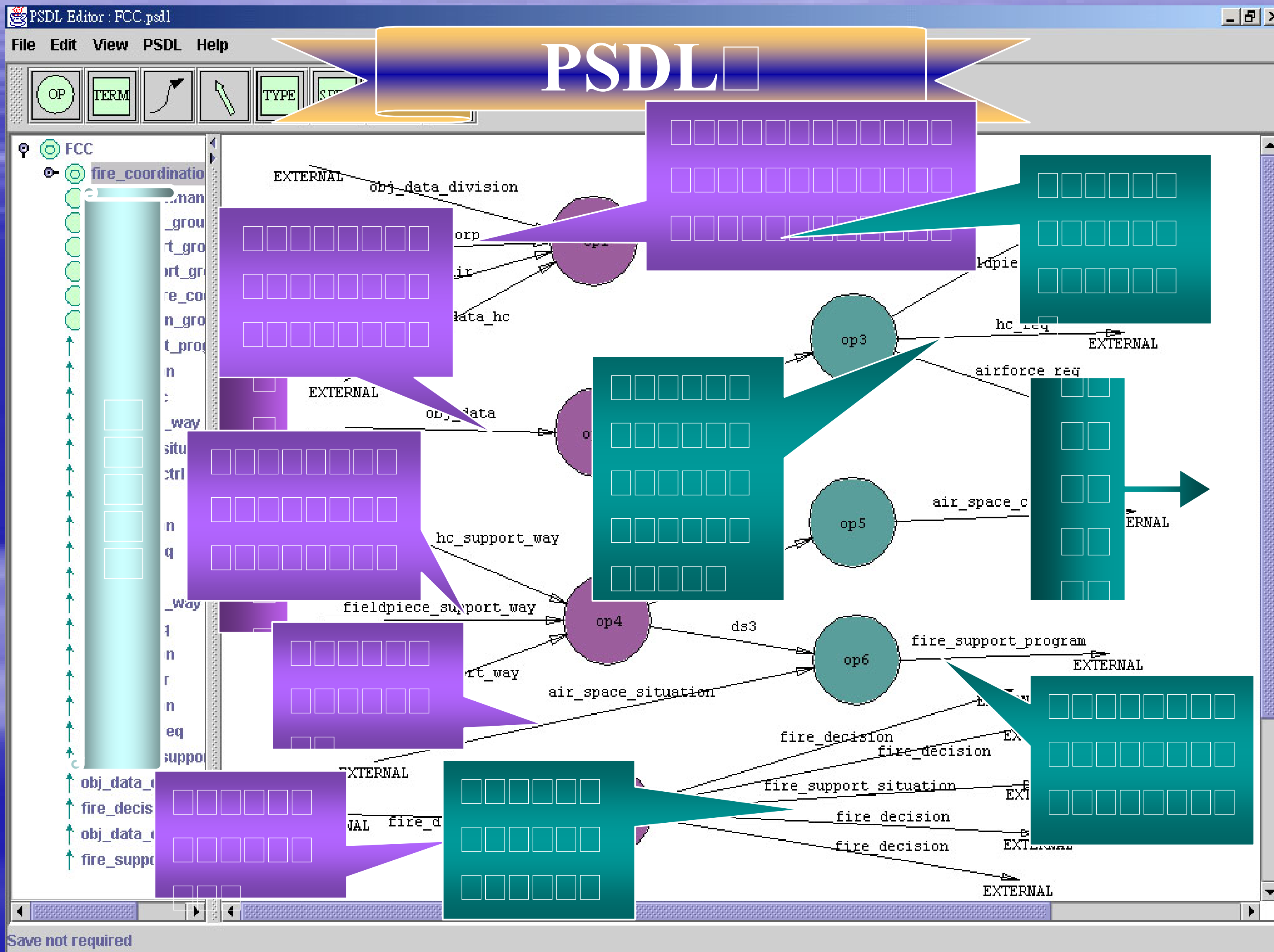
The Battlefield Objects of Fire Support and Coordination System (FSCS)

- Military People
  - *Fire-coordinating Officer, Fire-supporting Director, Artilleryman, Navy Liaison Officer, Air Force Liaison Officer, Chemistry Officer, Communication and Information Officer, Object-obtaining Officer, Object-analyzing Officer, Air Force Support Director, J2 Air Operational Officer, Air Control Director, Army Air Force Officer*
- Weapon Systems
  - *Field Artillery, Rocket Forces, Armor Forces, Gunnery, Fighter-bomber, Hunting Helicopter*
- Navigation Systems
  - CNS, Satellite Navigation GPS
- Platform Sensors
  - *Battlefield Surveillance and Reconnaissance Radar, Coast Acquisition Radar, Searching and Ranging Radar, Weather Radar, ATHS*
- Communication Links
  - *High-rate Receive Links, High-rate Transmit Links, Low-rate Receive and Transmit Links, Tadiran Communications*

Top-level of FSCS

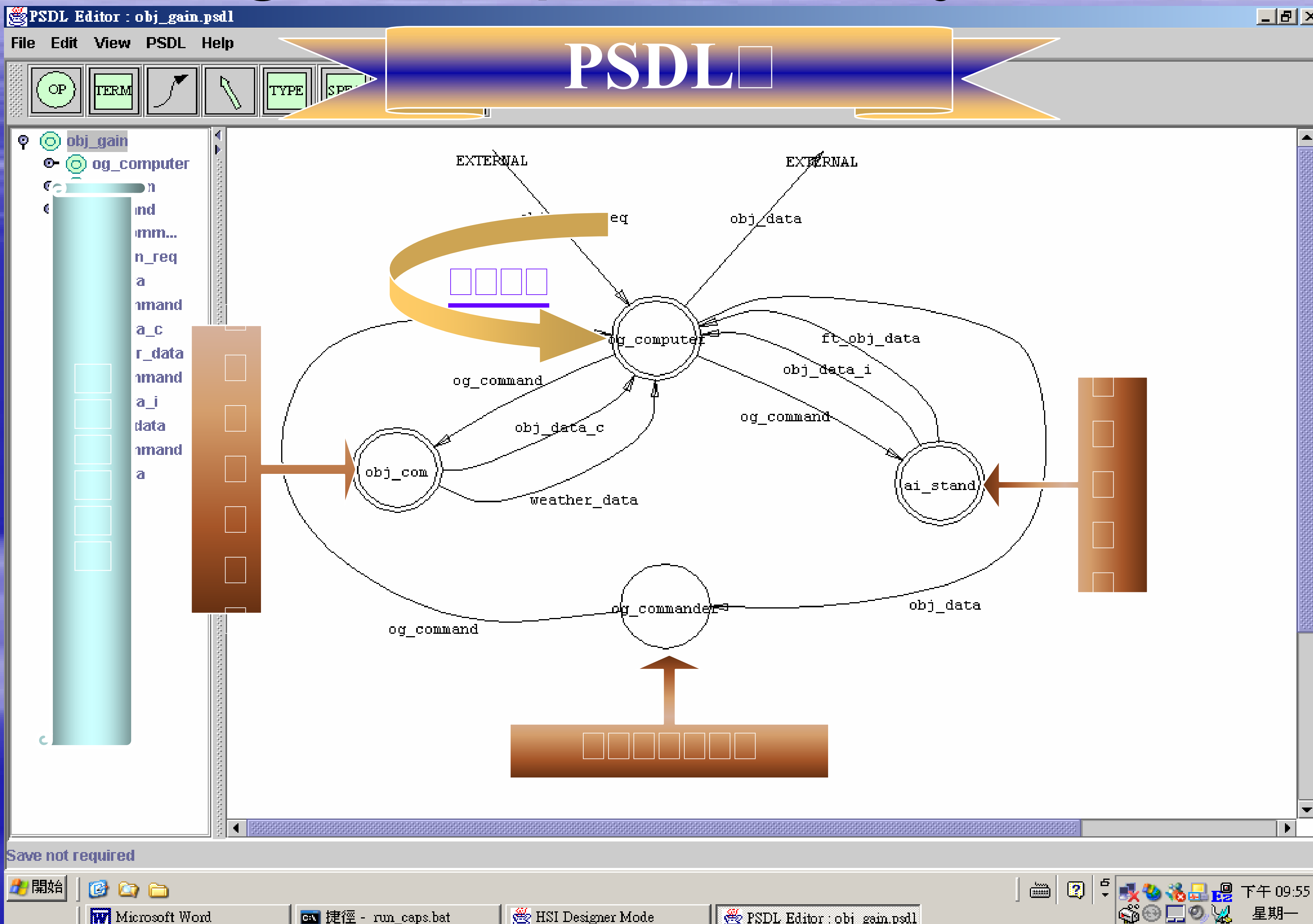


Mainframe

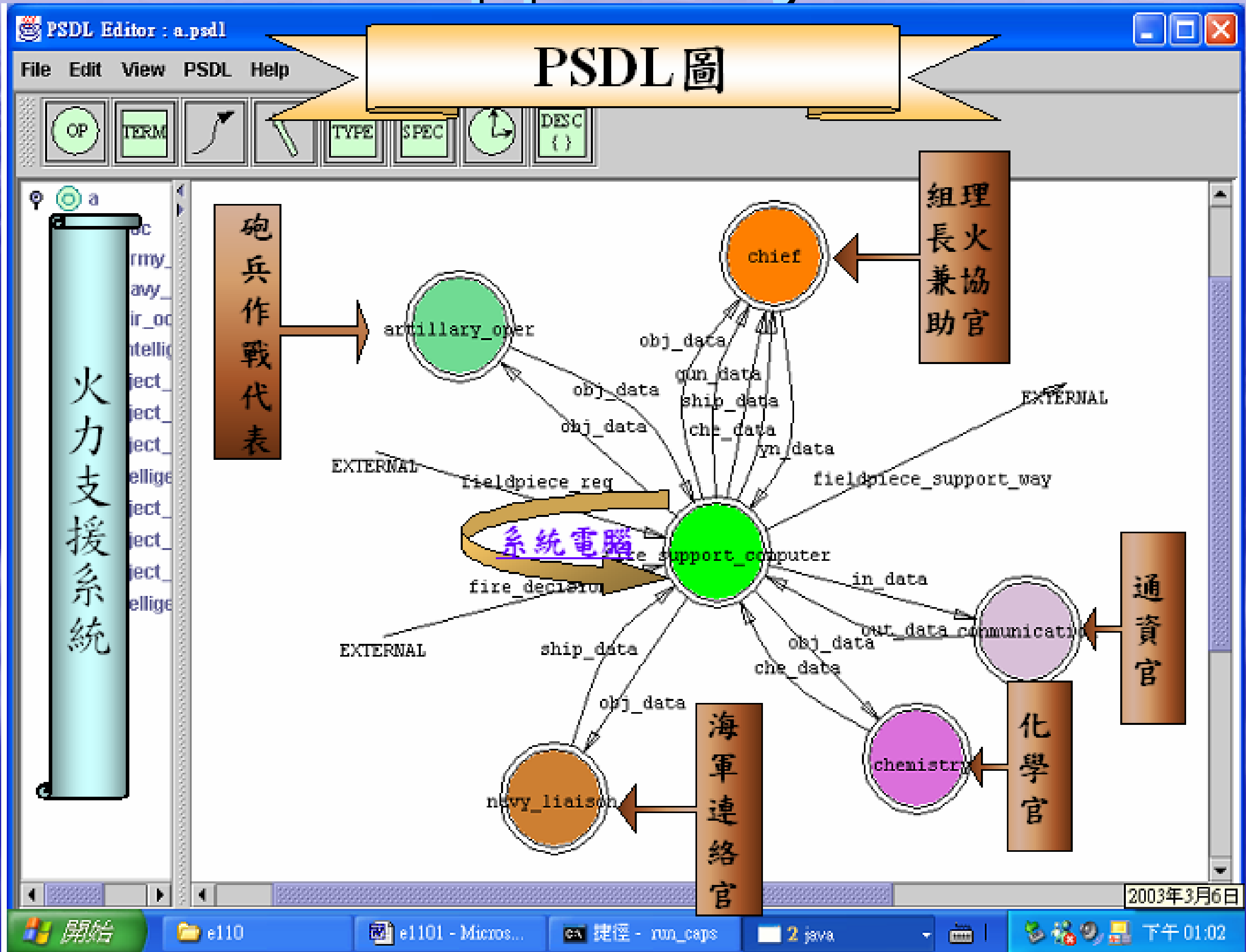


PSDL Editor : FCC.psdl
File Edit View PSDL Help
OP
TERM
TYPE
PSDL
FCC
fire_coordinatio
EXTERNAL
obj_data_division
op3
hc_req
EXTERNAL
airforce_req
obj_data
op5
air_space_c
ERNAL
hc_support_way
fieldpiece_support_way
op4
ds3
op6
fire_support_program
EXTERNAL
air_space_situation
fire_decision
fire_decision
fire_support_situation
fire_decision
fire_decision
EXTERNAL
EXTERNAL
obj_data_
fire_decis
obj_data_
fire_suppo
Save not required

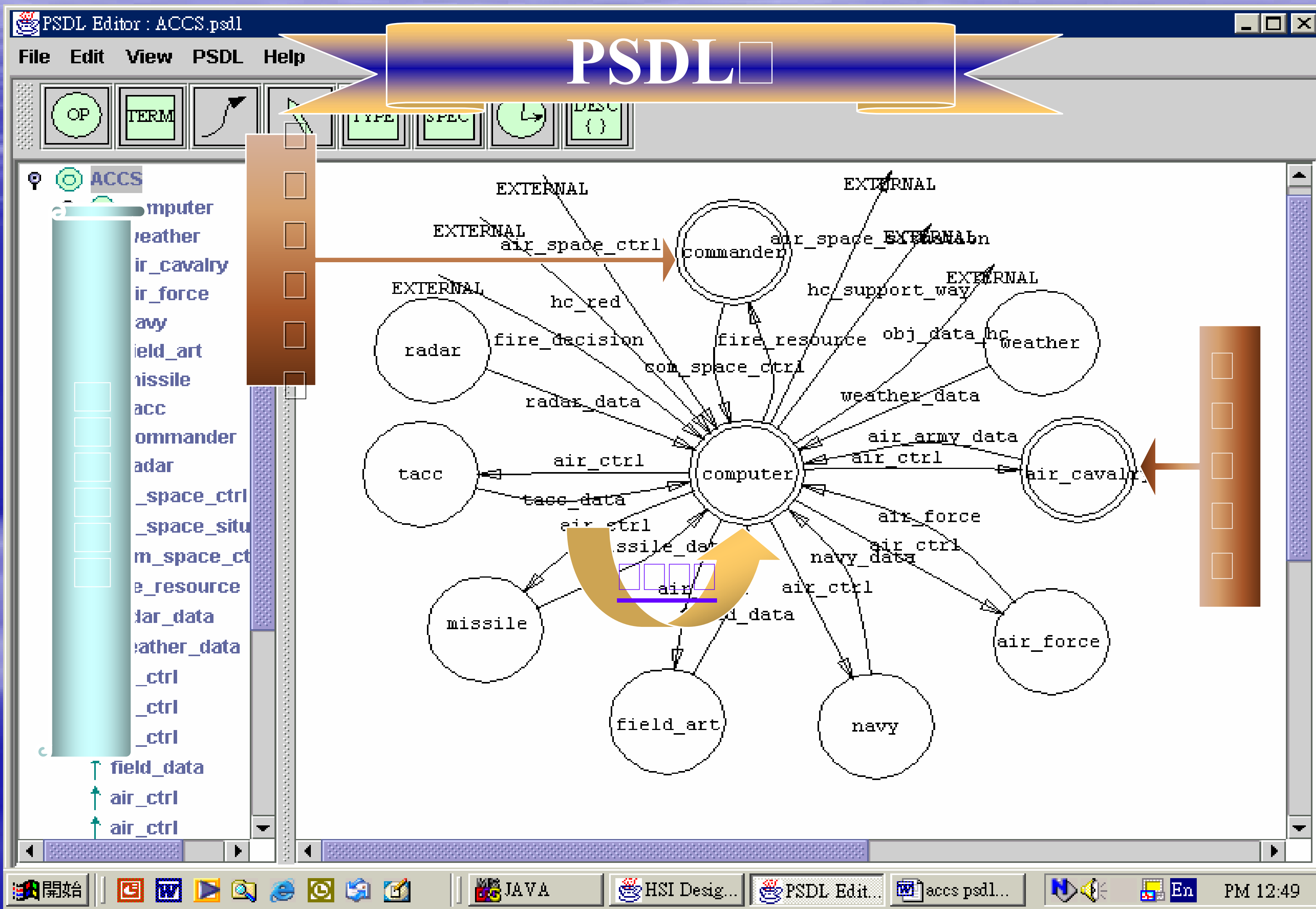
Target Acquisition System



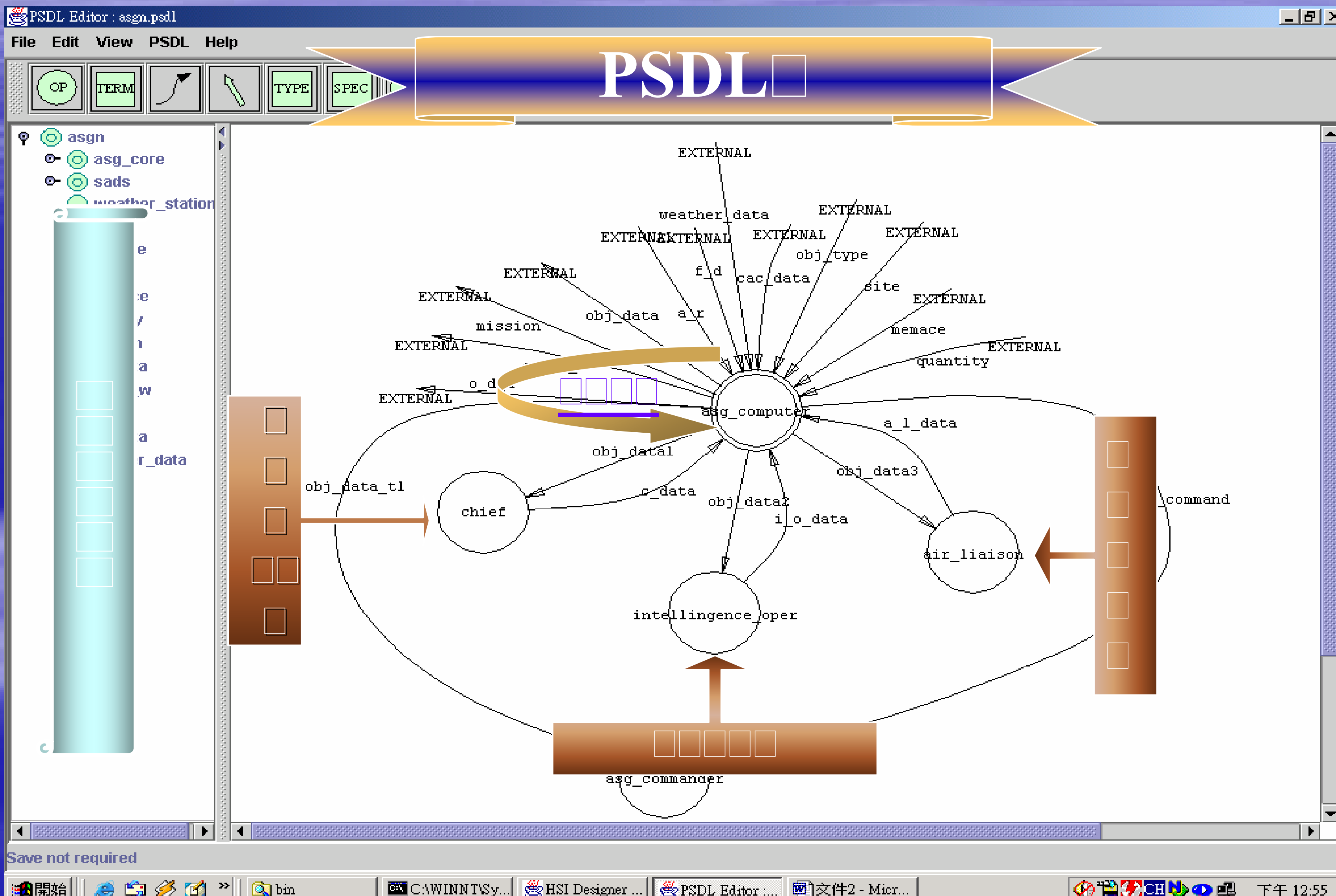
Fire Support System



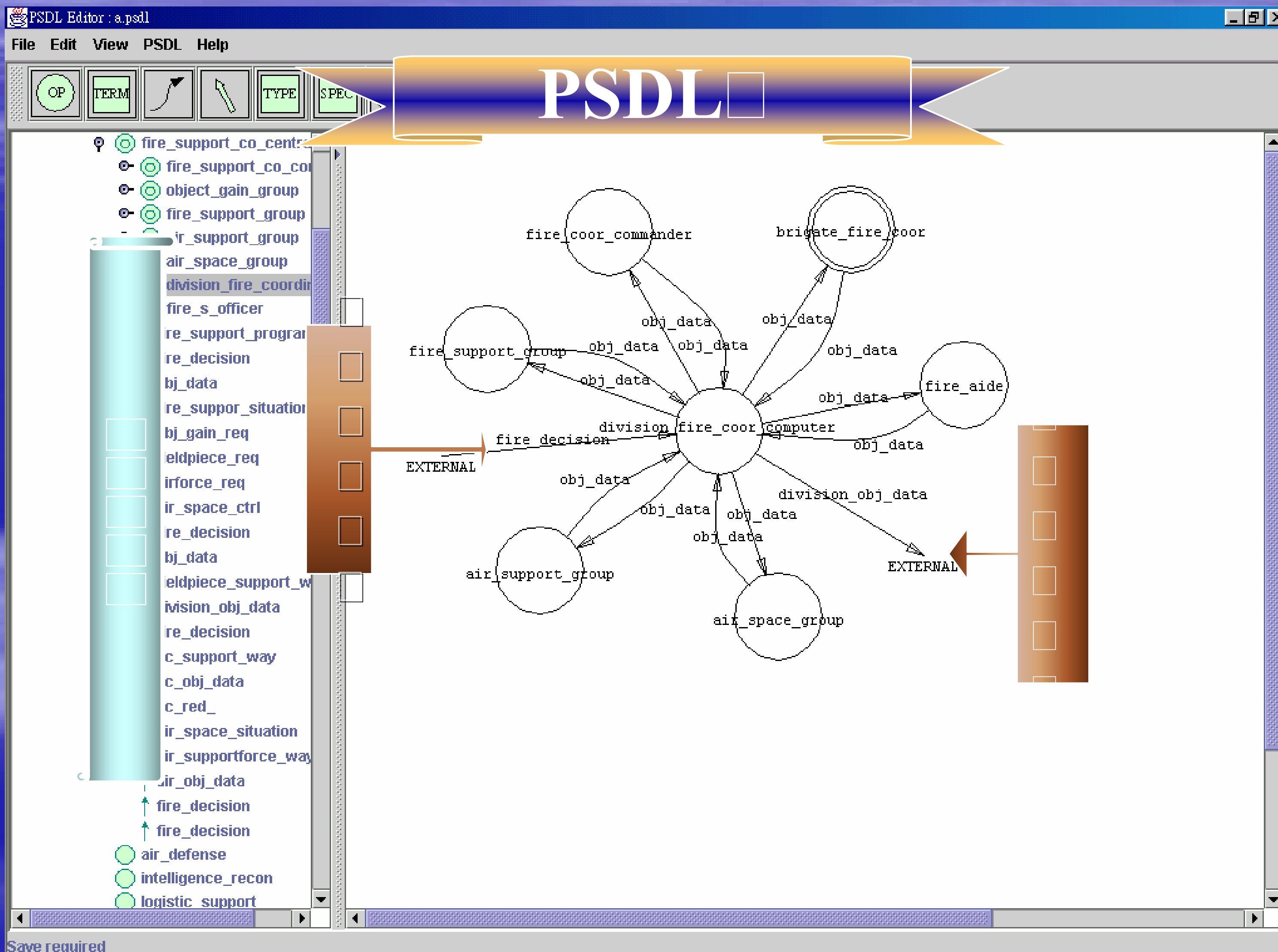
Air Control System



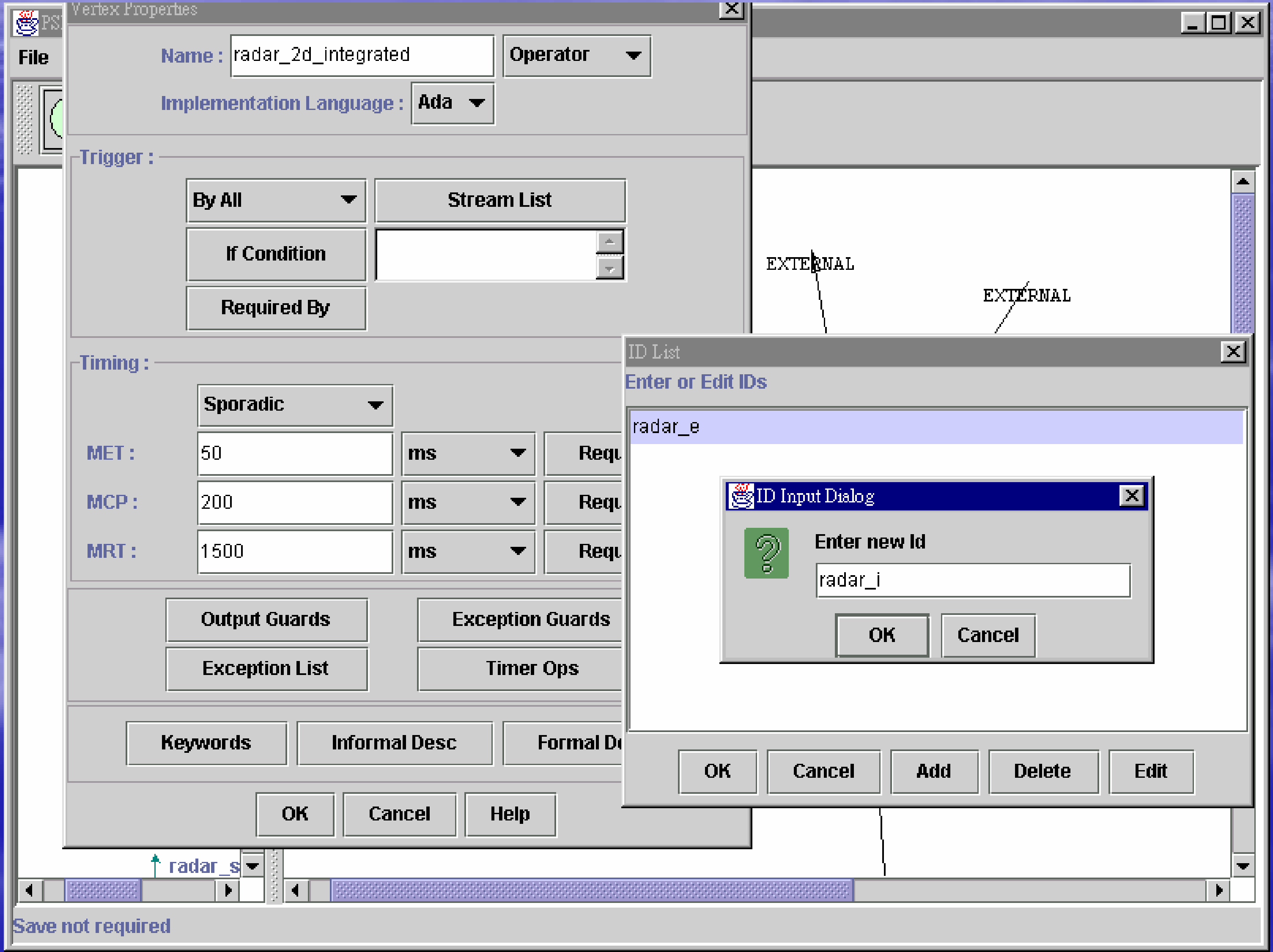
Air Force Support System



Fire Coordination System



Definition of Operator



The PSDL Code of FSCS

```
OPERATOR nlc_23
  SPECIFICATION
    INPUT radar : undefined_type
    INPUT object : undefined_type
    INPUT support  : support
    OUTPUT objectradar : undefined_type
    OUTPUT object : undefined_type
  END

  IMPLEMENTATION Ada nlc_23
  END

OPERATOR noc_26
  SPECIFICATION
    INPUT objectradar : undefined_type
    OUTPUT radar : undefined_type
    MAXIMUM EXECUTION TIME 0 ms
  END

  IMPLEMENTATION Ada noc_26
  END

OPERATOR sfcp_29
  SPECIFICATION
    INPUT object : undefined_type
    OUTPUT radar : undefined_type
  END

  IMPLEMENTATION Ada sfcp_29
  END
OPERATOR bnlo_32
  SPECIFICATION
    INPUT object : undefined_type
    OUTPUT object : undefined_type
  END
```

```
IMPLEMENTATION
    GRAPH
      VERTEX nlo_34_0
        PROPERTY x = 278
        PROPERTY y = 205
        PROPERTY radius = 35
        PROPERTY color = 62
        PROPERTY label_font = 5
        PROPERTY label_x_offset = 0
        PROPERTY label_y_offset = 0
        PROPERTY met_font = 5
        PROPERTY met_unit = 1
        PROPERTY met_x_offset = 0
        PROPERTY met_y_offset = -40
        PROPERTY is_terminator = false
      EDGE radar EXTERNAL -> nlo_34_0
        PROPERTY id = 39
        PROPERTY label_font = 5
        PROPERTY label_x_offset = 0
        PROPERTY label_y_offset = 0
        PROPERTY latency_font = 5
        PROPERTY latency_unit = 1
        PROPERTY latency_x_offset = 0
        PROPERTY latency_y_offset = -40
        PROPERTY spline = " 77 87 "
      EDGE object nlo_34_0 -> EXTERNAL
        PROPERTY id = 46
        PROPERTY label_font = 5
        PROPERTY label_x_offset = 0
        PROPERTY label_y_offset = 0
        PROPERTY latency_font = 5
        PROPERTY latency_unit = 1
        PROPERTY latency_x_offset = 0
        PROPERTY latency_y_offset = -40
        PROPERTY spline = " 254 23 "
    DATA STREAM
      radar : undefined_type,
      object : undefined_type
    CONTROL CONSTRAINTS
      OPERATOR nlo_34_0

   END
```

Lessons Learned

- Building methodologies and tools:
  - *IDEF*, *UML*, *C4ISR Architecture Framework 2.0*, *System Architect V9.1*, *ARIS*, *Rhapsody*
- Petri-Net
- *swap_into*, *swap_out_of*, *modified* and *unchanged* mechanism
- Nonmilitary C4ISR systems