

# High-Level System Design Using Foresight



## *Attention: Please Read This Page*

This paper was originally presented at the **OMG SBC Workshop** held in November of 2000. While we believe that the material is important and will be interesting to current users of Foresight Systems software, we have been through some changes since the paper was prepared.

Nu Thena Systems, Inc. and Foresight Systems, Inc. were predecessor companies to Foresight Systems M&S. All references to Nu Thena and Foresight Systems, Inc. are obsolete. Readers should assume that Foresight Systems M&S replaces all instances of Nu Thena Systems, Inc. and Foresight Systems, Inc.

The current contact information for Foresight Systems M&S is:

**Foresight Systems M&S**  
12550 N. 89th St.  
Scottsdale, AZ 85260

<http://www.foresightsystems-mands.com>  
[fs\\_marketing@foresightsystems-mands.com](mailto:fs_marketing@foresightsystems-mands.com)

The diagrams and illustrations included here were prepared with versions of the software current at the time that the presentation was prepared. Some of them may not be identical to equivalent images from the current version of the software. We apologize for any potential confusion.

Foresight's software is currently supported on the following computing platforms:

- Windows 2000/XP/Vista
- Linux (x86 based)
- Solaris V10

If you have questions, concerns, or issues, please contact:

Dean Stevens  
President, Commercial Systems Group  
Foresight Systems M&S  
[deans@foresightsystems-mands.com](mailto:deans@foresightsystems-mands.com)  
(408) 278-3983



# *High-Level System Design Using Foresight*

*Giovanna Di Marzo Serugendo*  
*IT / CE*

## *Outline*

---

- ◆ Development Process
- ◆ HW/SW Co-Design
- ◆ Foresight: a Modelling and Simulation tool
- ◆ ALICE DAQ System

# *Development Process (1)*

---

## ▶ Separate Functionality from Architecture

- Design system functionality before thinking at hardware/software implementation details

## ▶ Formal Specification and Verification

- Mathematical definition of system (unambiguous)
- Semantics of specification provides a model
- Behaviour of model = behaviour of system
- Verification: model behaves correctly (simulation, model checking)

# *Development Process (2)*

---

## ▶ First Phase: Functional Requirements

- **Abstract specification:** interfaces, functionality
- **Verification:** incomplete/inconsistent functional requirements, performance problems, design errors
- **Analysis:** critical parameters, maximum (minimum) performances, particular conditions

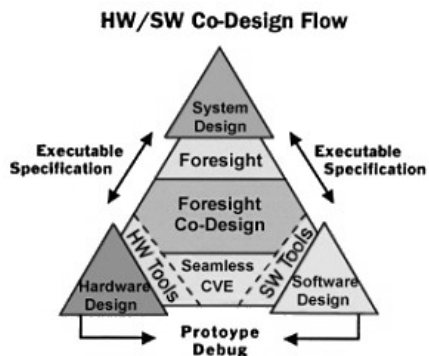
## ▶ Second Phase: Architectural Concerns

- **Detailed specifications:** algorithms, hardware choices, alternative architectures
- **Verification, analysis:** check requirements and performances

# *HW/SW Co-design*

---

- ▶ Foresight Systems, Inc.
- ▶ System Design
  - Foresight tool
  - Specification Execution
- ▶ System Co-design
  - Foresight co-design tool
  - Foresight specification with hw/sw components
  - Specification Execution
- ▶ HW/SW Components
  - HDL simulation environment
  - Seamless CVE



5

# *Foresight (1)*

---

- ▶ Foresight Tool
  - System Level Modeling and Simulation Tool
- ▶ Specification
  - Hierarchical Specifications
  - Data Flow Diagrams (event-driven processes, events, control)
  - State Transitions
  - Mini-specs
  - Real-time parameters

IT Seminar/November 2000

6

## *Foresight (2)*

---

### ► Analysis

- Type checking, input/output checking, syntax errors

### ► Execution of Specification (Simulation)

- Real-time execution of specification
- Stand-alone executable specification
- Animation of Diagrams
- Real-time constraint validation
- Debugging functions (breakpoints, monitors windows)
- Simulation is NOT formal verification !
- Works on Sun workstation

IT Seminar/November 2000

7

## *ALICE DAQ*

---

### ► Model of whole ALICE DAQ System

- Trigger System (L0, L1, L2)
- Trigger and Tracking Detectors
- DAQ (with sub-event building, event building, storage)
- Parameters (buffer sizes, etc.)

### ► Evaluation of Performances

- Whole system: maximal bandwidth / real bandwidth
- For each detector: buffer occupancy, bandwidth usage

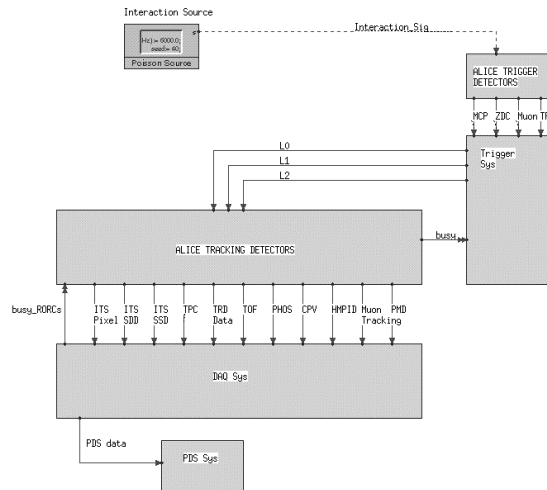
### ► Alternative Algorithms

- Event building computing
- L2 trigger decision

IT Seminar/November 2000

8

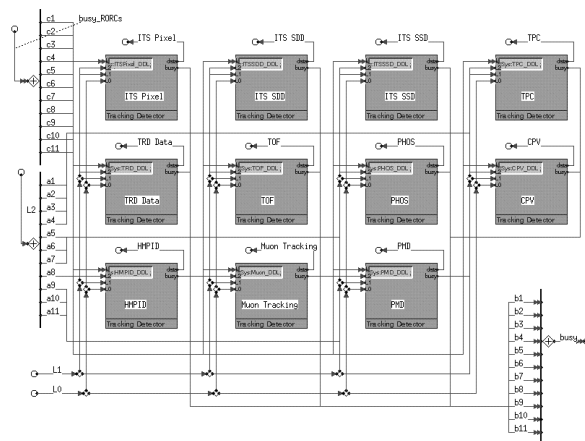
# ALICE: Overall System



IT Seminar/November 2000

9

# ALICE: Tracking Detectors



IT Seminar/November 2000

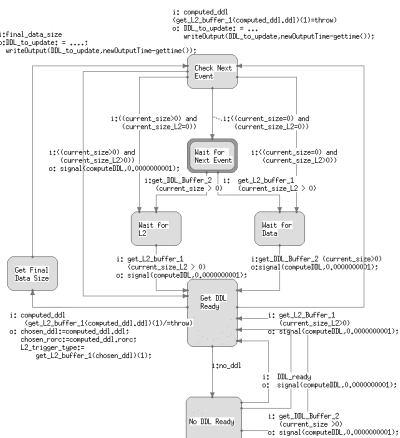
10

# ALICE: FSM, Mini-Spec

```

Inputs: DIL_ready, k, get_DIL_Buffer_2, get_L2_Buffer_1, current_size, current_size_L2,
final_data_size, computed_dil, no_dil, no_rorc;
Outputs: DIL_to_update, L2_trigger_type, computed_DIL, busy_dil;
Parameters: DIL_rate, thro_time, DIL_no;
Procedures: computeOutputTime(Outputs: OutputListType, OutputTimeType);
Static locals: chosen_dil, chosen_rorc, outputList;
Locals: newOutputList, newOutputTime, OutputTime;
Initialize:
o: outputList(1)=1;0;
END;

```



```

Inputs: data;
Outputs: bandwidth_rate;
Static locals: total_data_size;

Initialize:
BEGIN
total_data_size:=0;0;
END;

Procedure:
BEGIN
total_data_size:=total_data_size + data.data_type.size;
bandwidth_rate:=total_data_size/gettime();
END;

```

# Results

## Maximal Bandwidth

	L0	L1	L2
Central	137	133	92
Dimuon	462	457	585
Dielectron	159	152	197
Minbias	747	714	409
Misc	203	197	
Interaction	1997		

After 1 sec (6038 ev)  
Expected at L2:

	Buffer Full	Bandwidth	Maximum
TPC	23	14200 MB/	18000MB/
TRD	43	1627MB/	1800MB/

	C	MB	DM	DIEL	Total
L2	20 H	20 H	650 Hz	200 Hz	890 Hz

# *Conclusion*

---

- ▶ **Separate Functionality from Architecture**
- ▶ **Foresight Systems provides integrated tools**
  - Formal Specification and Execution
  - Seamless replacement of formal components by hardware/software components
  - <http://www.nuthena.com/>
- ▶ **Advantage**
  - Correct errors before implementation
  - Think about the functional level (correct interfaces)