Architectures
and Program Generation

CSE Research Review
Robert Balzer
March 12, 1996

Scalable Application Generation

Reference Requirements

Application Engineer

Inconsistency
Incompleteness

Binder

Application Requirements

Reference Architecture

System Generator
(Component Generator + Packager)

Application Engineer

External

Internal
Generic Structures
(e.g. Reference Requirements & Reference Architecture)

Declarative Workflow
Registers Model Interest
Detects Model Export
Notifies Interested Agents
Model Management Agents
Export Completed Models
Map into local representation
Import Available Models
Map from local representation
Locally Integrate Incoming Models
Integration is Rule Driven
Export Detected Errors
Message Handler Example

Field Data => Field Coordinator => Dataset Compiler

Dataset Definition
- Field Usage
  - Name
  - Type
  - Validations
  - Definition
  - Fields

(Extended) Field Usage
- Name
- Definition
- Type
- Validations

Field Definition
- Name
- Definition
- Type
- Validations
- Usage Type

Program Generator Infrastructure

Domain Specific Languages

Raise the Ceiling
- Transformation Based Compilation
- Executable Specifications
- Translator

Raise the Foundation
- Programming Language
- Executable Specification Language
- mailbox
Three Basic Environment Levels

- Application Engineer
- End User
- Domain Architect

Program Generation automates development tasks within the constrained variability of a family

- Generate Components
- Generate Glue between Components
- Adapt Components to Architecture Evolution
Replenishment at Sea
Naval Convoy Resupply

Layout RAS + L4 + R4

NOT TO SCALE
Further activities not specified in the diagram:

- Upon arrival at RF fueling station, hook up, fuel, and then breakdown.
- Abort emergency? do emergency-breakdown(ə RF).

Prevent undesirable overlaps:

- These events must not overlap; reassignments will be delayed to assure this:
  - First minute of departures approaches and 1st minute thereafter.
  - Overlap depart-approach: (S1,RF->S1,LF; S2,RF->S2,LF; S3,LF->S3,RF; S4,RF->S4,LF + 1 minute).

Action definitions:

- Simulate hook up of a ship at a station by waiting 20 minutes.
- Definition: hook up(ə ship) = wait 20 minutes.
- Definition: fuel(ə ship) = wait \((0.95 \times (fuel-capacity(ə ship) - fuel-remaining(ə ship))) / fuel-rate(ə ship))\.
- Assert fueled?(ə ship) = simulate filling to 95% capacity.
- Definition: breakdown(ə ship) = wait 15 minutes; assert breakdown-completed(ə ship).
- Definition: emergency-breakdown(ə ship) = wait 5 minutes; assert breakdown-completed(ə ship).

Predicate definitions:

- Definition: left-side-all-fueled? = (every left-side-stations, s, is fueled(ə asg(s))).
- Definition: right-side-all-fueled? = (every right-side-stations, s, is fueled(ə asg(s))).

True if every ship assigned to a left (or right) side station is fueled.

Scenario Generation

- Commanded Scenario = Synthetic Generation

Training

Identify Live Units

Synthetic Forces

Event Actuated

Constrained Response

Orders
Program Generation automates development tasks within the constrained variability of a family

- Generate Components
- Generate Glue between Components
- Adapt Components to Architecture Evolution

Architecture Connectors

Conduit for all inter-module interactions

- Network Sockets
- Event Broadcast
- Corba
- RPC

Inserted Mediators enable

- Instrumentation
- Interface adaptation
- Filtering
- Value Added Infrastructure

Uniform Mediator
Interface Spanning
Integration Frameworks
Semi-Transparent Interfaces

- **Transparent API**
  - Useable by Unmodified Applications

- **Augmented API**
  - Provides Value-Added Capabilities
  - Sources of Control
    - Static "Resource" Configuration Files
    - Dynamic Third-Party Controller

---

**Caching Connector**

Legend

- Transparent API
- Augmented API
Caching Connector
Augmented API

- What to Cache
  - Matching Client Request with Server Response
- Cache Integrity
  - Is a Cached object still valid
    - Always
    - Time Duration or Observable Event
    - Query to Determine
- Cache Retention Policy
  - Flush Least Recently Used
  - Retain/Flush Selector
- Cache Allocation
  - Initial size
  - Dynamic Growth

Augmented Infrastructure
Through Indirect Invocation
Architecture Infrastructure for Inter-Module Interaction

- Provide infrastructure for managing and manipulating inter-module connectors
  - (All inter-module interactions occur through these connectors)
  - dynamic probes - instrument & monitor behavior
  - redirect or alter messages, spawn reactive processes
  - redirect or alter responses, spawn reactive processes
  - move events from one integration space to another
- Allow others to provide middle-ware services based on this infrastructure

NSWC GeoServer

- Calculate which targets are in which regions
- Determine Response
  - Operator Alerts
  - ID Targets
  - Issue Engagement Orders
- Determine Radar Coverage Adequacy
Testbed Architecture

Doctrine Groups
Rules
Activations

NYU

GeoServer

Intersections

Display Server

Doctrine Validation

Coverage Map

SWC

Map

Track Server

Tracks

Actions

ISI

Doctrine Authoring

NYU

Tracks

ISI

Alert: Engaging N7666 with weapons from ENGAGE.
Simulated Demo

Self Guided Demonstration Service
Step 1: Record Scenario
Self Guided Demonstration Service

Step 2: Annotate Scenario

Knowledgeable User Annotated Scenario

Service

Self Guided Demonstration Service

Step 3: Novice Plays Back Scenario

Playback Modes:
- Single Step
- Auto Advance
- Exploration

Annotated Scenario
Adapt Components to Architecture Evolution

- Problem: Component Assumes Context
- Solution
  - Separate Component & Context Specifications
  - Generate Code from combined Specifications
  - Delay Generation until Context is determined
Exception Adaptation

Exception Identification
Object(s) involved
Causal Predicate
Default Repair(s)

Decontextualized Component Spec.

Abort
Repair Object
Return New Object
Accept Damaged Object

e.g. On Fuel-Exhausted \(\Rightarrow\) ship.speed=0

Exception Response

Adapt Components to Architecture

Partial Path Expressions
- Exception Identification
- Interface Specification

Decontextualized Component Spec.

Context
- Specification

Target Selection
- Exception Response

Complete Object Class Hierarchy

Adapted Component

ISI
Future Challenges

- Architecture as declarative target
- Client Maintenance
- Coverage

Product Line Development
Use Appropriate Design Notations
Product Line Development

Challenge: Coverage

Programming Languages

Specification Languages

Problem Oriented Languages

Product Line Development

Coverage Challenge: Broader Languages

Programming Languages

Specification Languages

Problem Oriented Languages
Product Line Development
Coverage Challenge: More Languages

Generating Architecturally Compliant Components