

# Visually representing data-driven analysis using state diagrams

**Simon Delisle**

**Michel Dagenais**

---

Progress Report Meeting  
May 2015



**POLYTECHNIQUE  
MONTRÉAL**

LE GÉNIE  
EN PREMIÈRE CLASSE

# Presentation plan

- Introduction
- Related Work
- Architecture
- Tool presentation
- Applications and use cases
- Future work

# Introduction

- Trace analysis to find performance issues
- Different tools available (Jumpshot, Trace Compass)
- Force users to use specific analysis and trace type

- Trace Compass → Data-driven analysis with XML specification
- User-friendly capture of all the data-driven trace analysis information

# Related Work

- Trace analysis
- Languages
  - DSL
- Modeling
  - Papyrus
  - Frameworks

# Trace analysis

- State system method
  - Build a state machine from trace event
  - Often uses an hard-coded state provider
- States inside traces
  - SLOG2 and Jumpshot
  - Good technique to show information in Gantt views
  - Limited to a trace type

# Languages

- Declarative language
  - Snort
  - Easily modeled
- Imperative language
  - DTrace and SystemTap
  - No graphical representation
- Automata-based language
  - STATL

# Languages

- Domain specific language (DSL)
  - Declarative language
  - Solve domain specific problems
  - Benefits
    - Domain expert can understand, validate and develop DSL programs
  - Disadvantages
    - Need to maintain the languages
    - User must learn a new language



# Papyrus

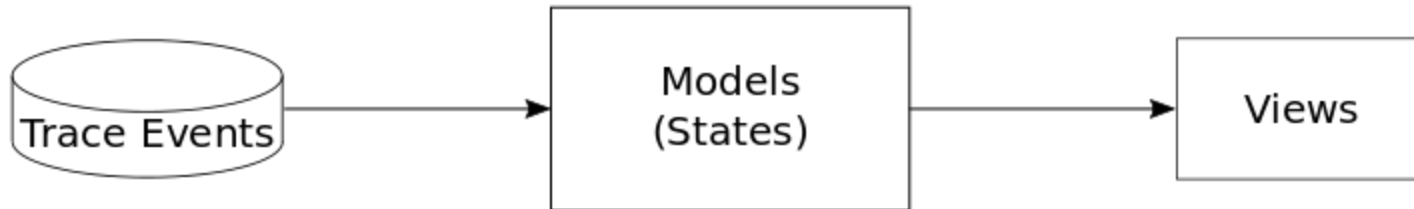
- Modeling tool
- UML support
- Eclipse based
- Create DSL

# Modeling Frameworks

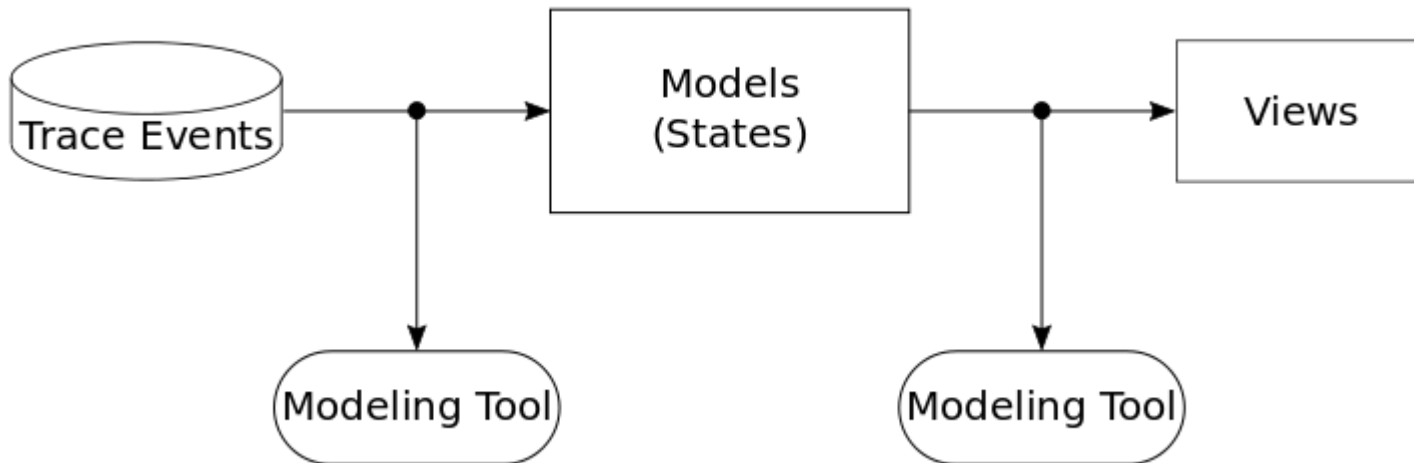
- **GMF (Graphical Modeling Framework)**
  - Used to build other graphical tools
- **Graphiti**
  - Relatively new
  - GMF alternative
  - Easy to use and learn
- **Sirius**
  - Built on top of GMF
  - No programming to build our tool

# Architecture

- Old fashioned

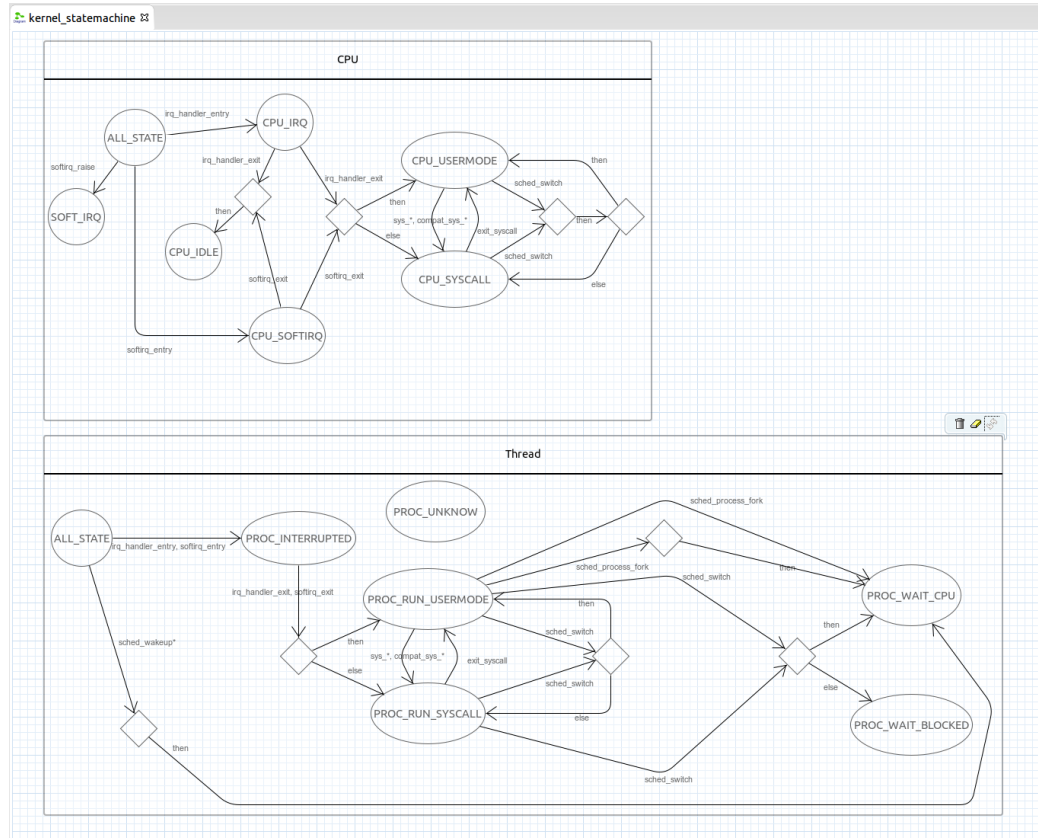


- New way



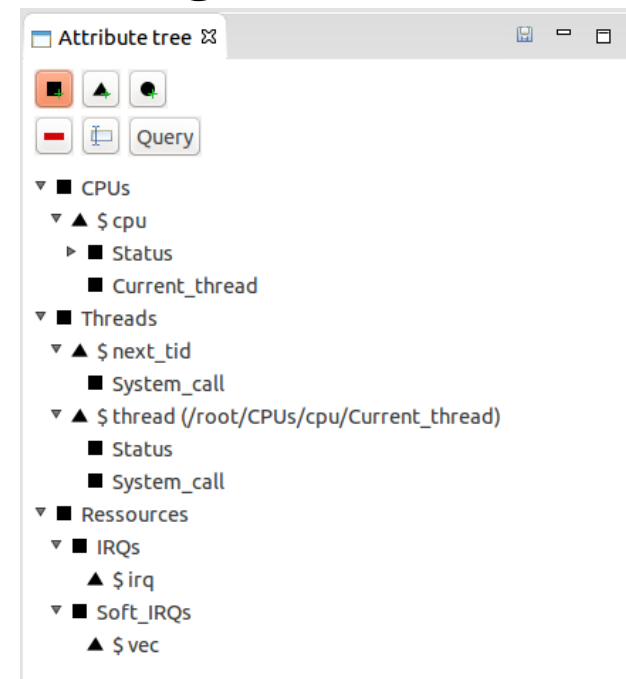
# State machine

- Based on UML
- Pseudostate
  - Initial
  - Condition
- States
- Transition
  - State changes



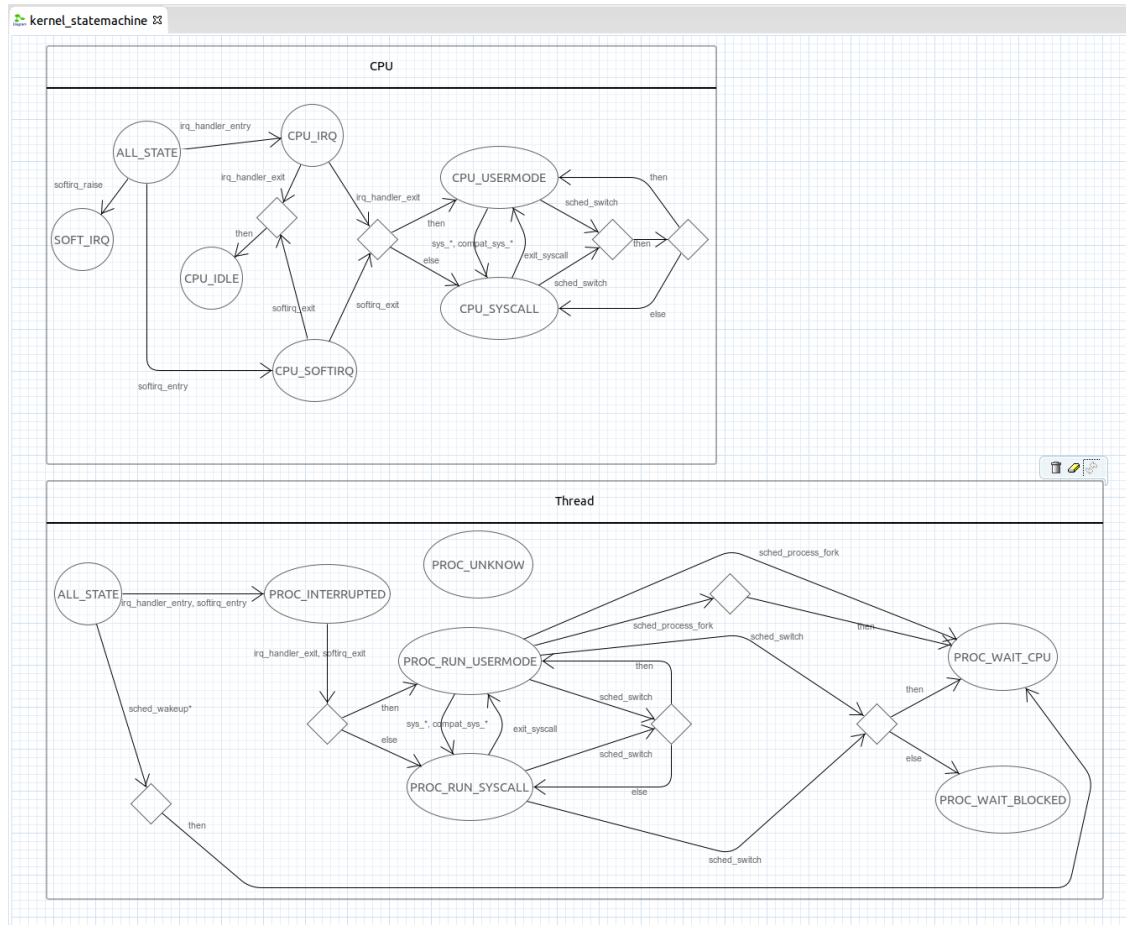
# Attribute Tree

- Tree based editor
- Used in the state machine editor
  - Specify the attribute that will be changed with the transition
- Can be reused



# Applications and Use Cases

- Kernel state machine



# Applications and Use Cases

- Analyse Trace Compass itself
  - Bad requests

Control Flow Resources RequestAnalysis

### Request Analysis

```
graph TD; CREATED((CREATED)) --- COALESCED((COALESCED)); COALESCED --- SUSPENDED((SUSPENDED)); SUSPENDED --- RUNNING((RUNNING)); RUNNING --- COMPLETED((COMPLETED));
```

Palette

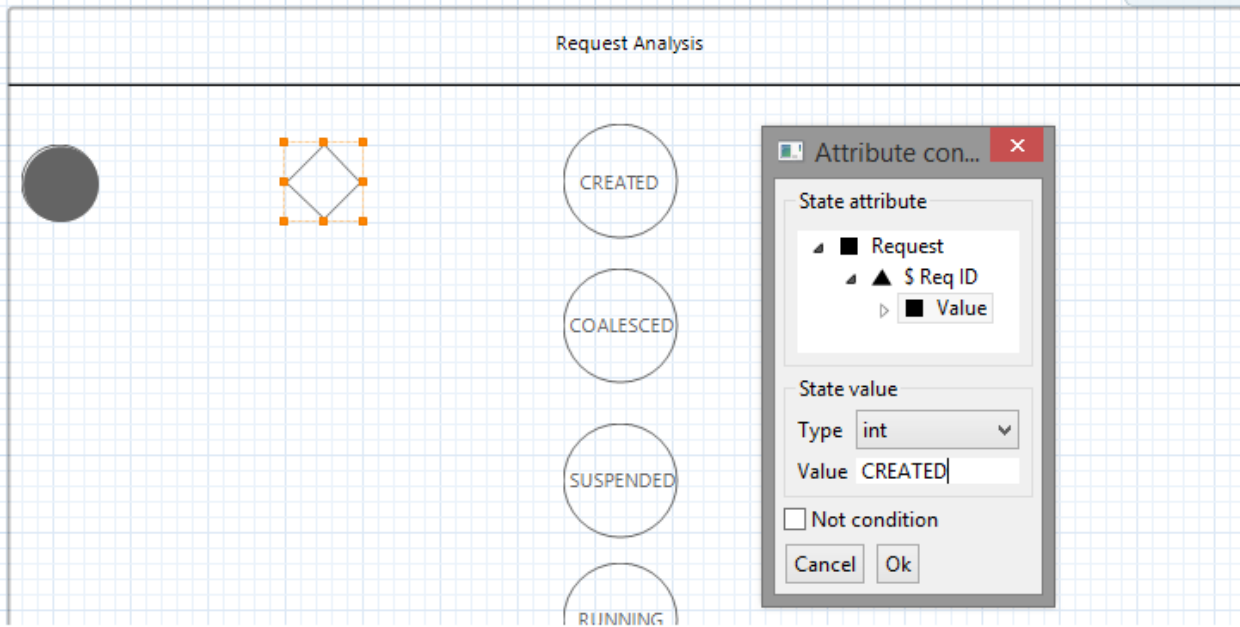
- Select
- Marquee
- Connections
- Transition
- Objects
  - State
  - Initial State
  - Final State
  - Condition
  - State machine

Properties TmfTrace.log RequestAnalysisTree (C:\Users\Simon\runtime-Tracecompass\Tracing\Statemachine\Tree\RequestAnalysisTree.attributetree)

Request

- ▲ \$ Req ID
  - ▲ Value
    - CREATED
    - COALESCED
    - SUSPENDED
    - RUNNING
    - COMPLETED
    - BAD





**Attribute con...**

State attribute

- Request
  - \$ Req ID
    - Value

State value

Type: int

Value: CREATED

Not condition

Cancel Ok

**Palette**

- Select
- Marquee
- Connections
- Transition
- Objects
  - State
  - Initial State
  - Final State
  - Condition
  - State machine

**Main**

Name	Condition

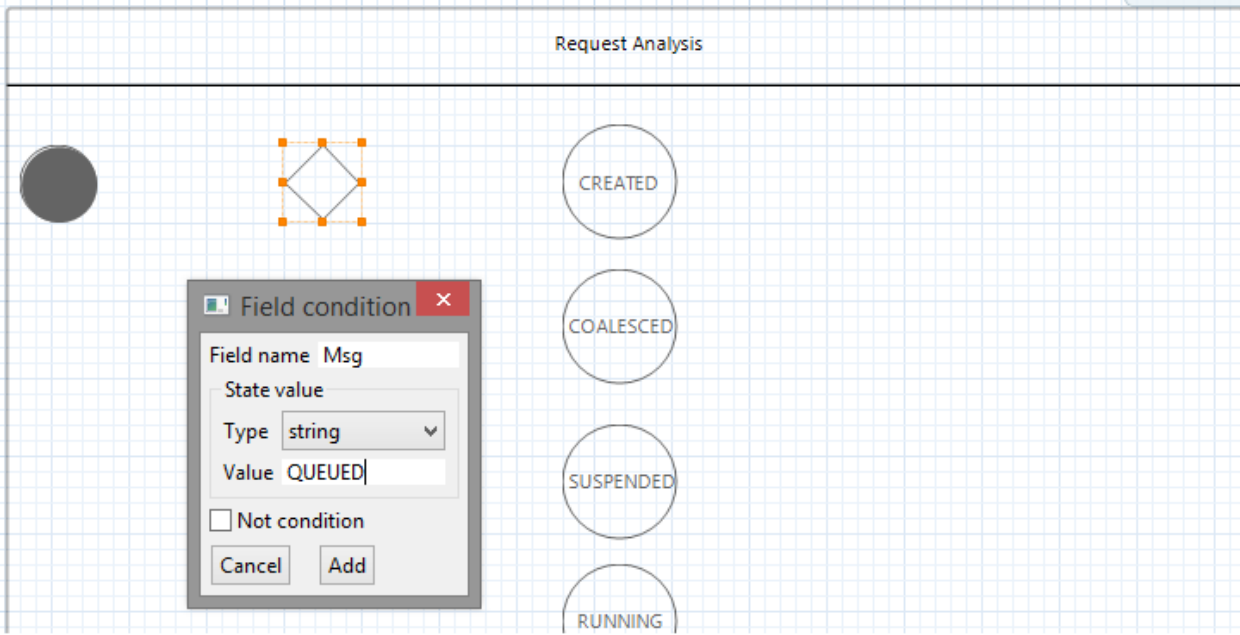
Add field condition

Add condition with attribute

Remove condition

Expression type

AND  OR



**Field condition** [X]

Field name: Msg

State value

Type: string

Value: QUEUED

Not condition

Cancel Add

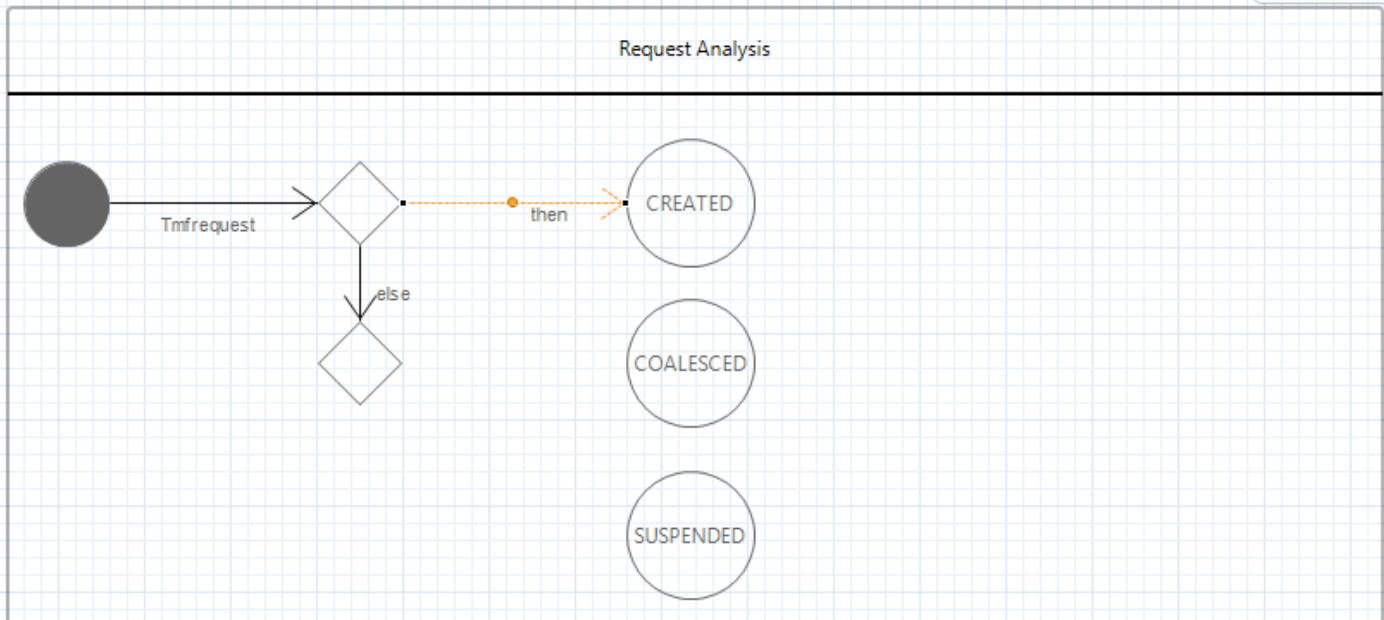
**Palette**

- Select
- Marquee
- Connections
- Transition
- Objects
  - State
  - Initial State
  - Final State
  - Condition
  - State machine

Main	Name	Condition
	/Request/Req ID/Value = CREATED	

Expression type:  AND  OR

Buttons: Add field condition, Add condition with attribute, Remove condition



**Palette**

- Select
- Marquee
- Connections
- Transition
- Objects
  - State
  - Initial State
  - Final State
  - Condition
  - State machine

**Main** Name: then

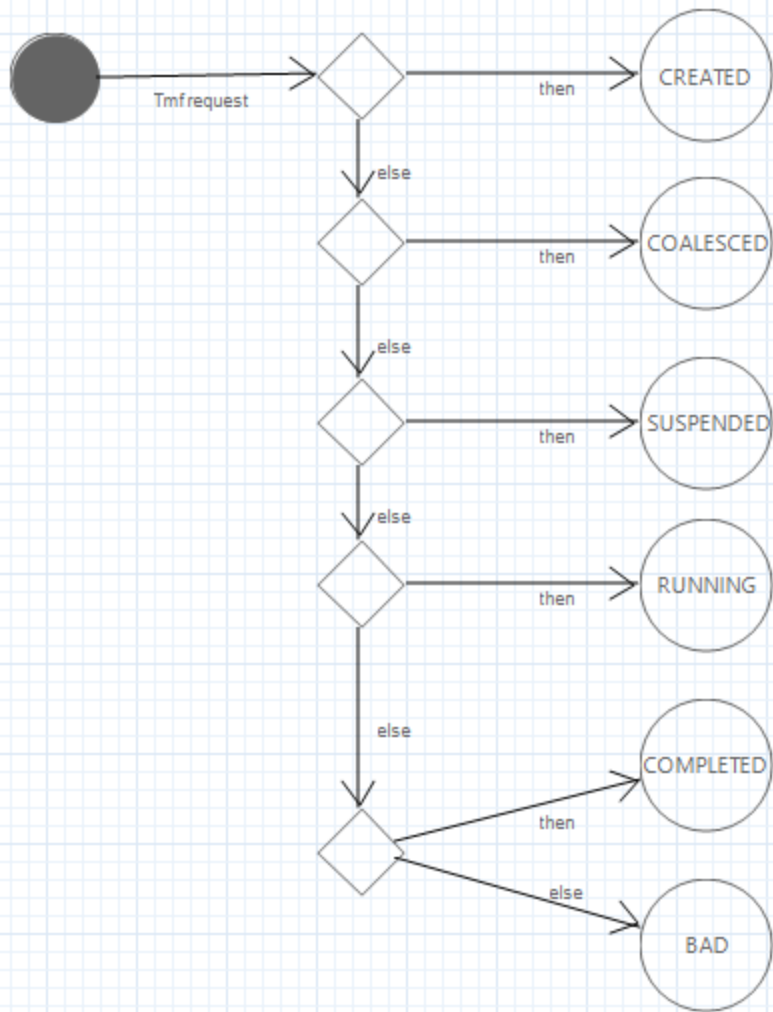
State Change

/Request/Req ID/Value = CREATED

Add/Edit state change

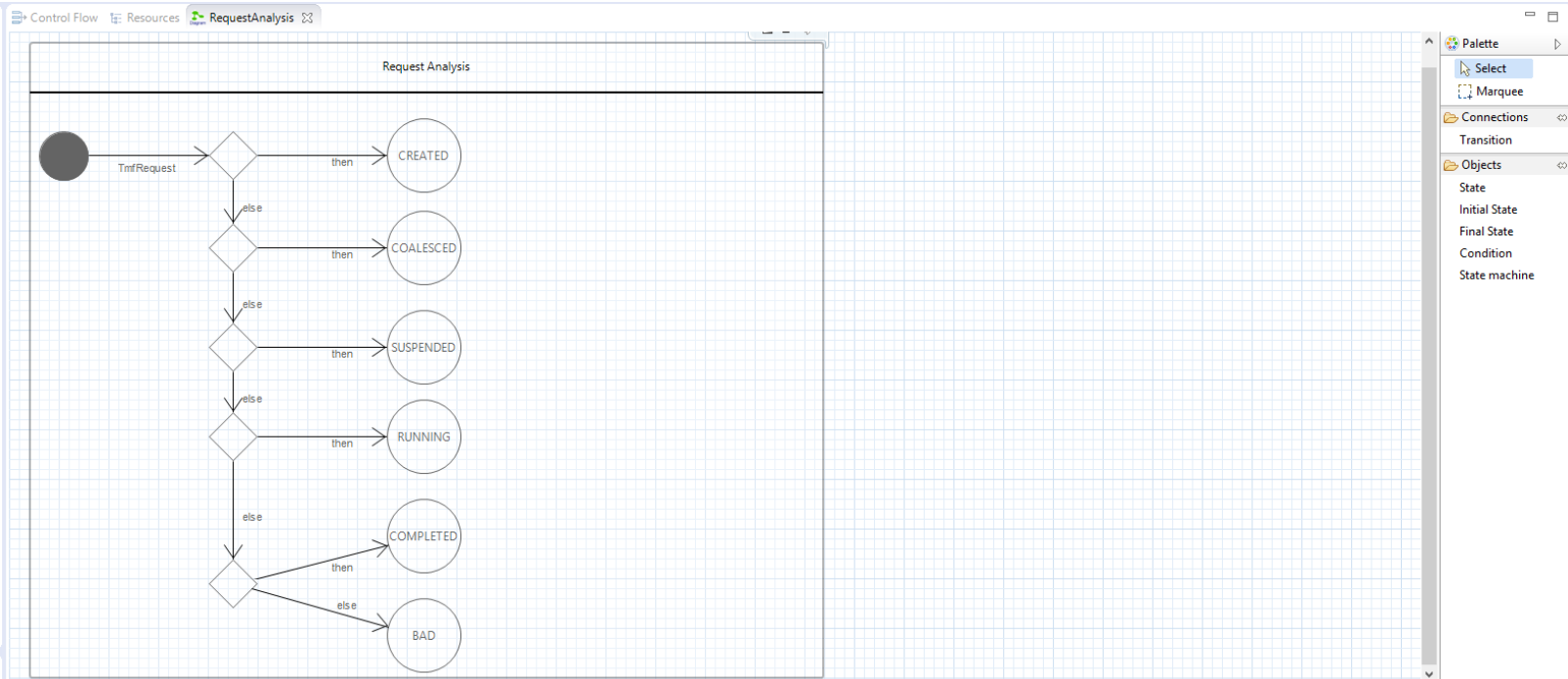
Remove state change

### Request Analysis



Project Explorer

- Tracing
  - Experiments [0]
  - Traces [1]
    - TmfTrace.log
      - Request Analysis
        - Request Analysis View
        - Test a builtin XML module file
        - Tmf Statistics Analysis
- Statemachine
  - Diagrams
    - RequestAnalysis.diagram
  - Tree
    - RequestAnalysisTree.attributetree



Palette

- Select
- Marquee
- Connections
- Transition
- Objects
- State
  - Initial State
  - Final State
  - Condition
  - State machine

Control

Histogram Bookmarks Properties TmfTrace.log RequestAnalysisTree (C:\Users\Simon\run-time-Tracecompass\Tracing\Statemachine\Tree\RequestAnalysisTree.attributetree) Request Analysis View

Name	ID	ParentID
TmfTrace.log	TmfTrace.log	
0	0	
1	1	
2	2	
3	3	
4	4	
5	5	

# Future Work

- Synchronisation between views
- Define filter and pattern
- Better handling of all files that are used to defined analysis

# Conclusion

- We have a complete editor to create analysis
- A more efficient way to create a state provider