

# APPLICATION OF SOFTWARE HEALTH MANAGEMENT TECHNIQUES

NASA Cooperative Agreement NNX08AY49A

Nagabhushan Mahadevan

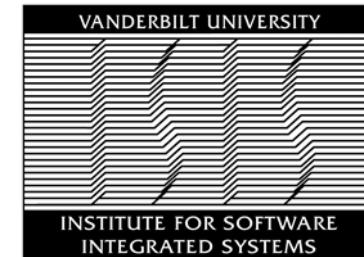
Abhishek Dubey \*

Gabor Karsai

Institute for Software Integrated Systems

Vanderbilt University

Nashville TN



# Software Failure: Malaysian Air (Boeing 777) in-flight upset (2005)

- Airplane's autopilot experienced excessive acceleration values.
- Autopilot pitched nose-up to 17.6 degree and climbed at a vertical speed of 10,650 fpm.
- Airspeed reduced to 241 knots and aircraft descended 4,000 ft.
- Re-engagement of autopilot followed by another climb of 2,000 ft.

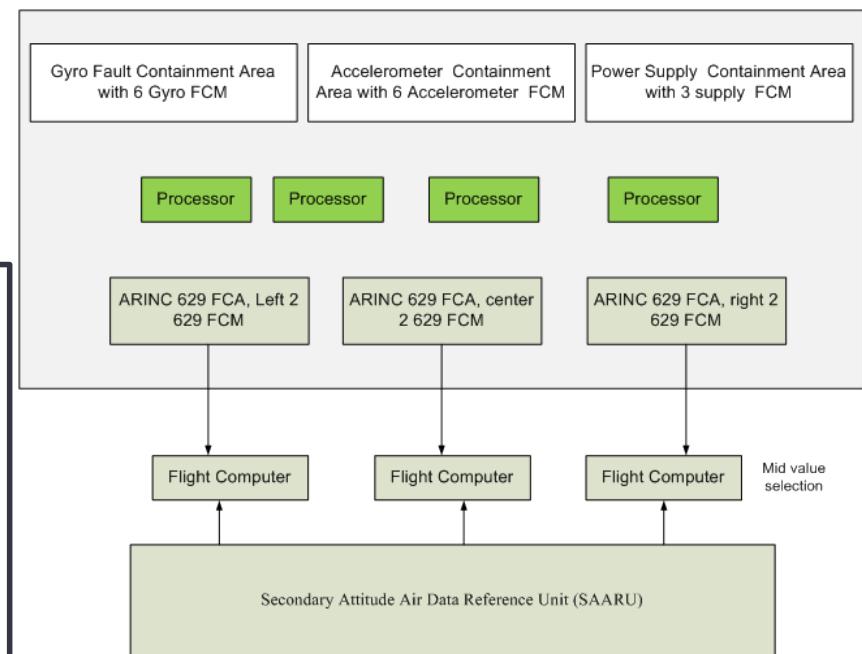
Contributing Factors: “An anomaly existed in the component software hierarchy that allowed inputs from a known faulty accelerometer to be processed by the air data inertial reference unit (ADIRU) and used by the primary flight computer, autopilot and other aircraft systems.”

-- from

<http://aviation-safety.net/database/record.php?id=20050801-1>



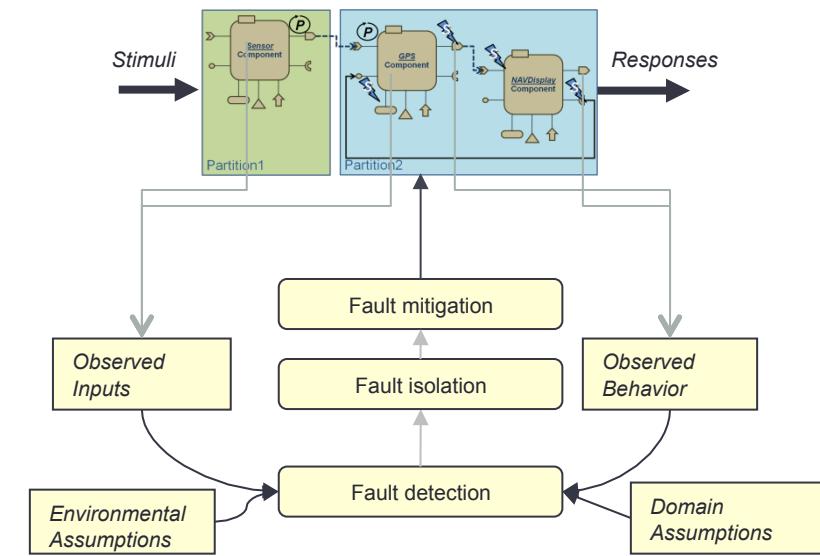
Source: Google Images



Based on Air Data Inertial Reference Unit (ADIRU) Architecture (ATSB, 2007, p.5)

# Software Health Management

- Software is a complex engineering artifact.
  - Software can have latent faults.
  - Faults appear during operation when unforeseen modes or interactions happen.
  - Techniques like Voting and Self-Checking pairs have shortcomings
    - Common mode faults
    - Fault cascades
- SHM is the extension of FDIR techniques in Physical systems to Software.

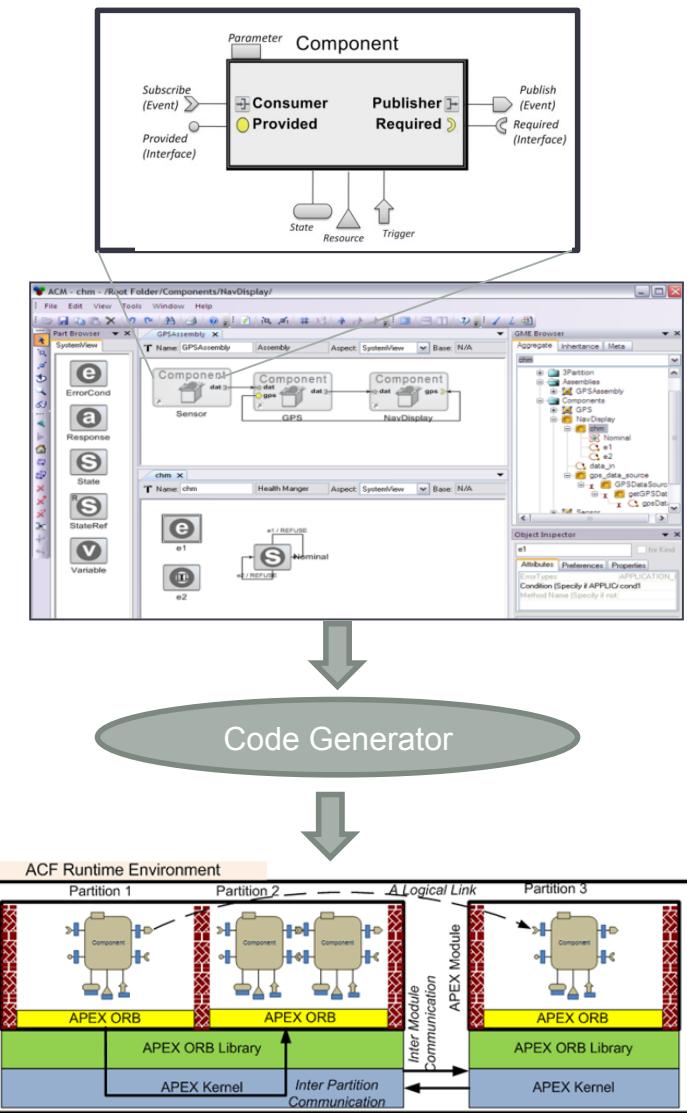


# OUR APPROACH

---

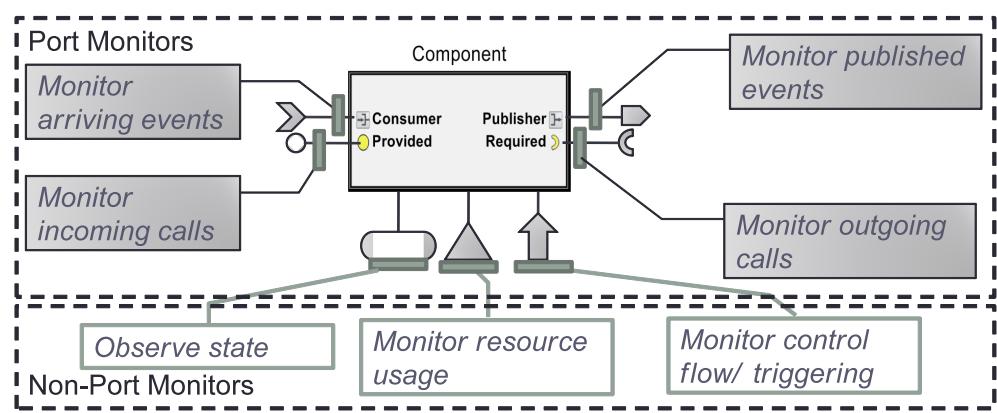
# System Structure

- Component-based system.
  - Strictly Specified Interactions
    - Synchronous Interfaces: call/return
      - Periodic: time-triggered
      - Aperiodic: event-triggered
    - Asynchronous Interfaces: publish-subscribe
      - Periodic: time-triggered
      - Aperiodic: event-triggered
- **Model-Based Software Development**
  - Specification of Monitoring expressions
  - Specification of Reactive Mitigation
- Hard-Real time ARINC-653 based runtime
  - Spatial separation
  - Temporal separation
  - Deadline Monitoring



# Anomaly Detection

- Model-Based Specification of Monitoring expressions
  - Post/Pre condition violations: threshold, rate, custom filter (moving average)
  - Resource Violations: Deadline
  - Validity Violation: Stale data on a consumer
  - Concurrency Violations: Lock Time Outs.
  - User code violations: reported error conditions from application code.
- Code Generators
  - Synthesize code for implementing this monitors in the system.

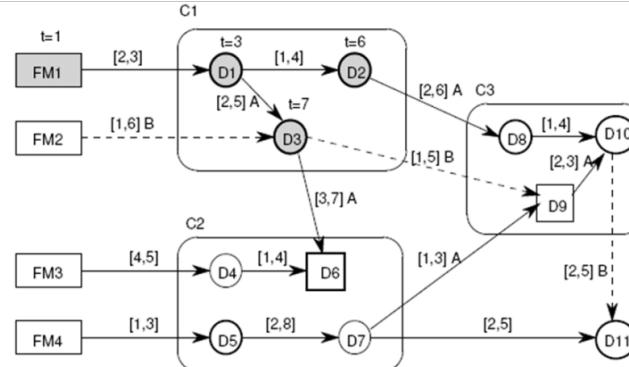


- Based on these local detection, each component developer can implement a local health manager
- It is a reactive timed state machine with pre specified actions.
- All alarms, actions are reported to the system health manager

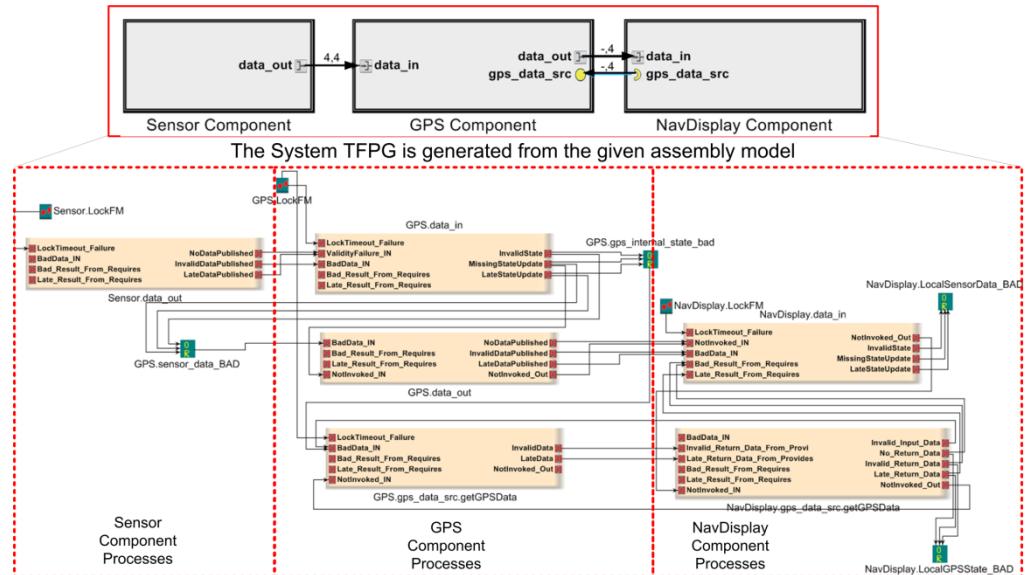
# Failure Diagnosis (Isolation)

- Model-based diagnoser is automatically generated
- Specification of data and control flow inside each component.
- Code Generators
  - Generate Failure Propagation graphs based on the system assembly and deployment information.
  - We use fault propagation templates developed beforehand for each category of component interaction.
- Overall, this gives a full timed failure propagation graph of the system.
- This graph is used to generate (multiple root) failure hypothesis that can explain the alarms seen in the system.
- Hypothesis are ranked by
  - Plausibility
  - Robustness

A TFPG model ( $t = 10$ , Mode=A  $\forall t \in [0, 10]$ )



Example: GPS Assembly

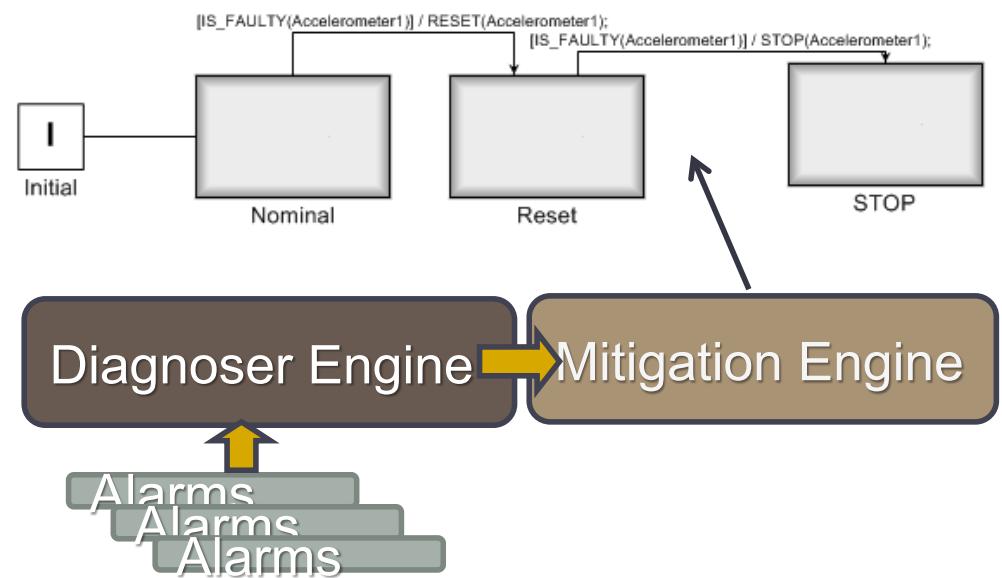


# System-level Fault Mitigation

- Model-based mitigation specification at two levels
  - Component level: quick action
  - System level: Reactive action taking the system state into consideration
  - System designer specifies them as a parallel timed state machine.
- Fixed set of mitigation actions are available
- Runtime code is generated from models
- **Advantages:**
  - Models are higher-level programs to specify (potentially complex) behavior
    - more readable and comprehensible
  - Models lend themselves to formal analysis – e.g. model checking

## List of predefined Mitigation Actions

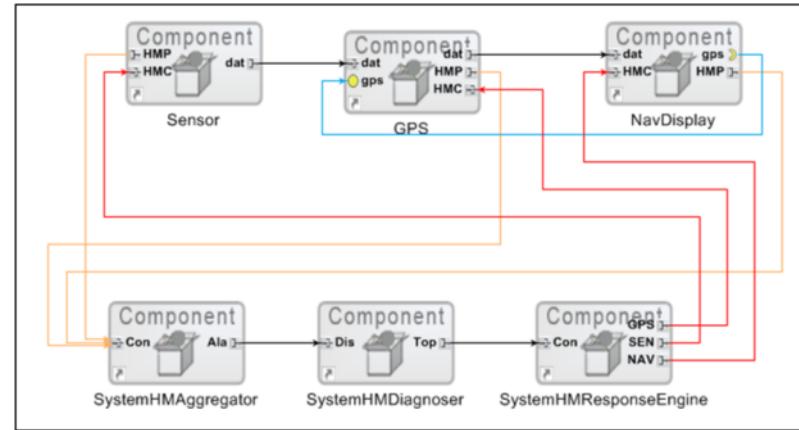
CLHM:ABORT	Discontinue current operation, but operation can run again
CLHM: USE PAST DATA	Use most recent data (only for operations that expect fresh data)
CLHM: STOP	Discontinue current operation
	Aperiodic processes (ports): operation can run again
	Periodic processes (ports): operation must be enabled by a future START HM action
CLHM: START	Re-enable a STOP-ped periodic operation
CLHM RESTART	A Macro for STOP followed by a START for the current operation
SLHM: RESET	Stop all operations, initialize state of components, start all periodic operations
SLHM: STOP	Stop all operations



# System-level Health Management: Data Flow

- 1. Aggregator:
  - Integrates (collates) health information coming from components (typically in one hyperperiod)
- 2. Diagnoser:
  - Performs fault diagnosis, based on the fault propagation graph model
  - Ranks hypotheses
  - Component that appears in all hypotheses is chosen for mitigation
- 3. Response Engine:
  - Issues mitigation actions to components based on diagnosis results
  - Based on a state machine model that maps diagnostic results to mitigation actions

These are automatically generated



*The Health Management flow::*

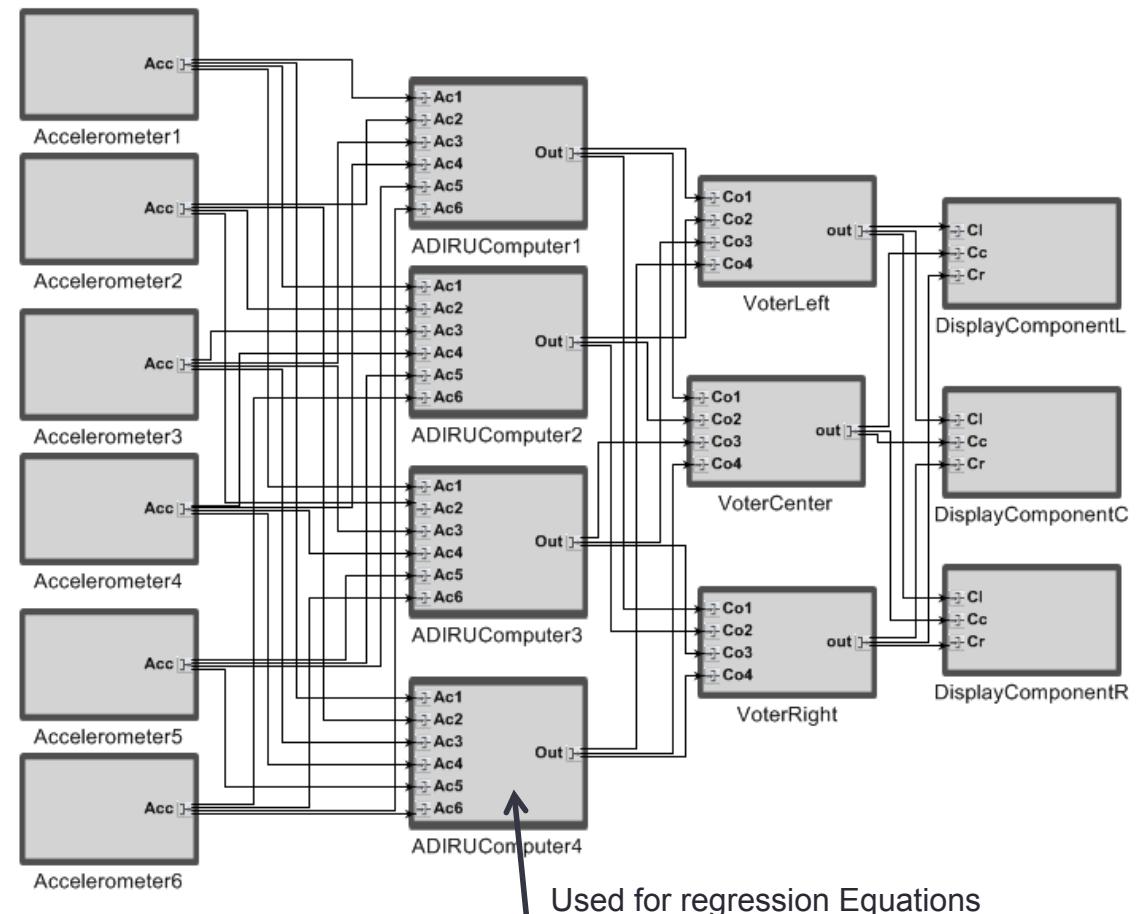
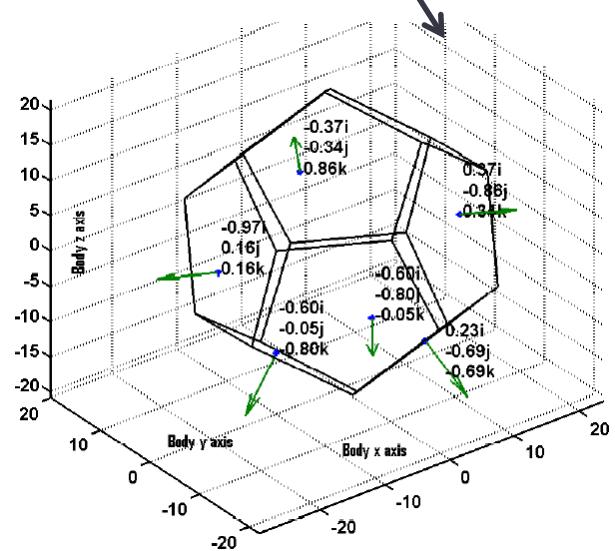
1. *Locally detected anomalies are mitigated locally first. – Quick reactive response.*
2. *Anomalies and local mitigation actions are reported to the system level.*
3. *Aggregated reports are subjected to diagnosis, potentially followed by a system-level mitigation action.*
4. *System-level response commands are propagated to components.*

# CASE STUDY

---

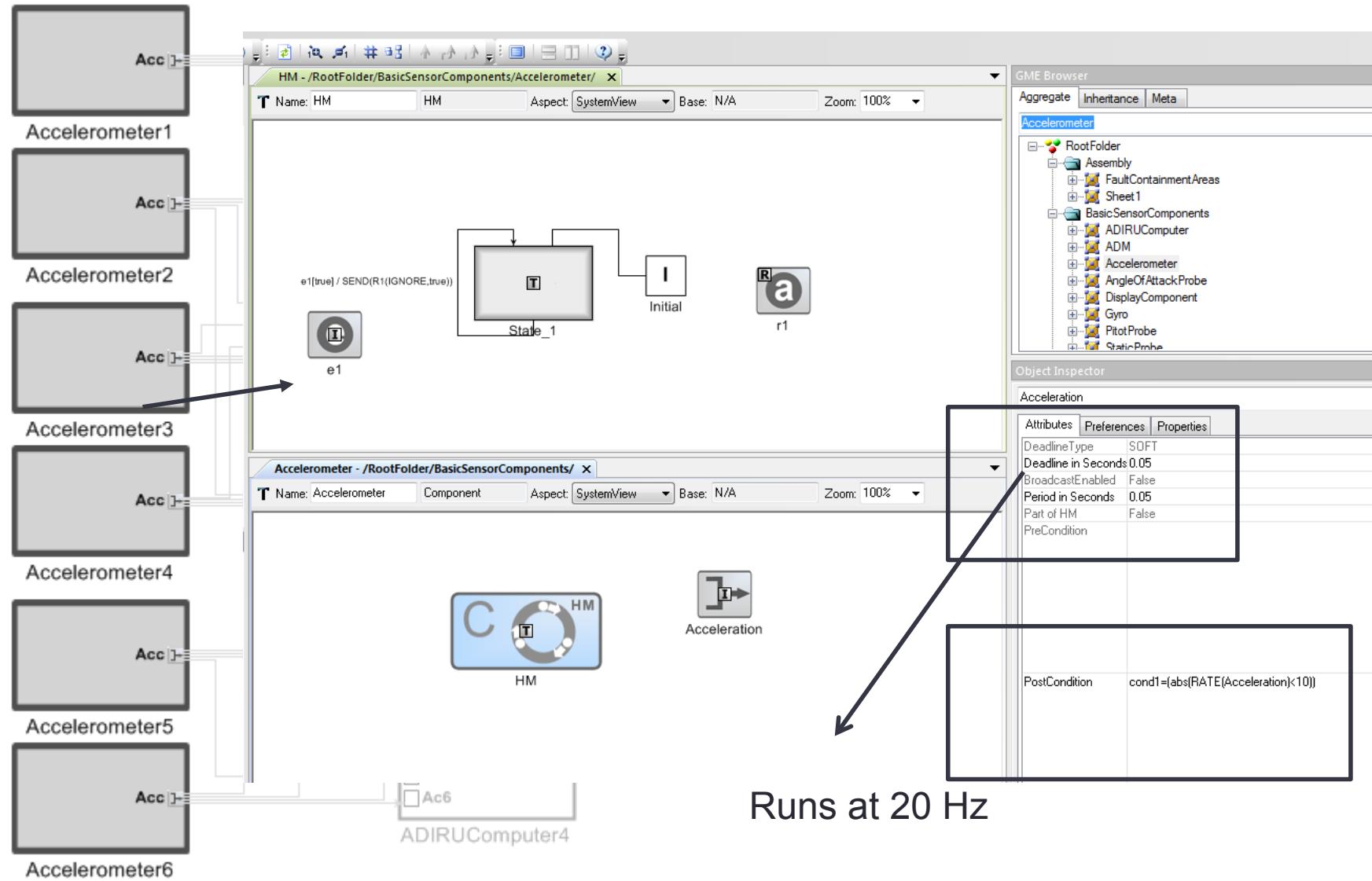
# Case Study

- Modeled the architecture as a software component assembly
- Created the fault scenario
- Only modeled part of the system to illustrate the point of SHM
- Accelerometers are arranged on six faces of a dodecahedron.



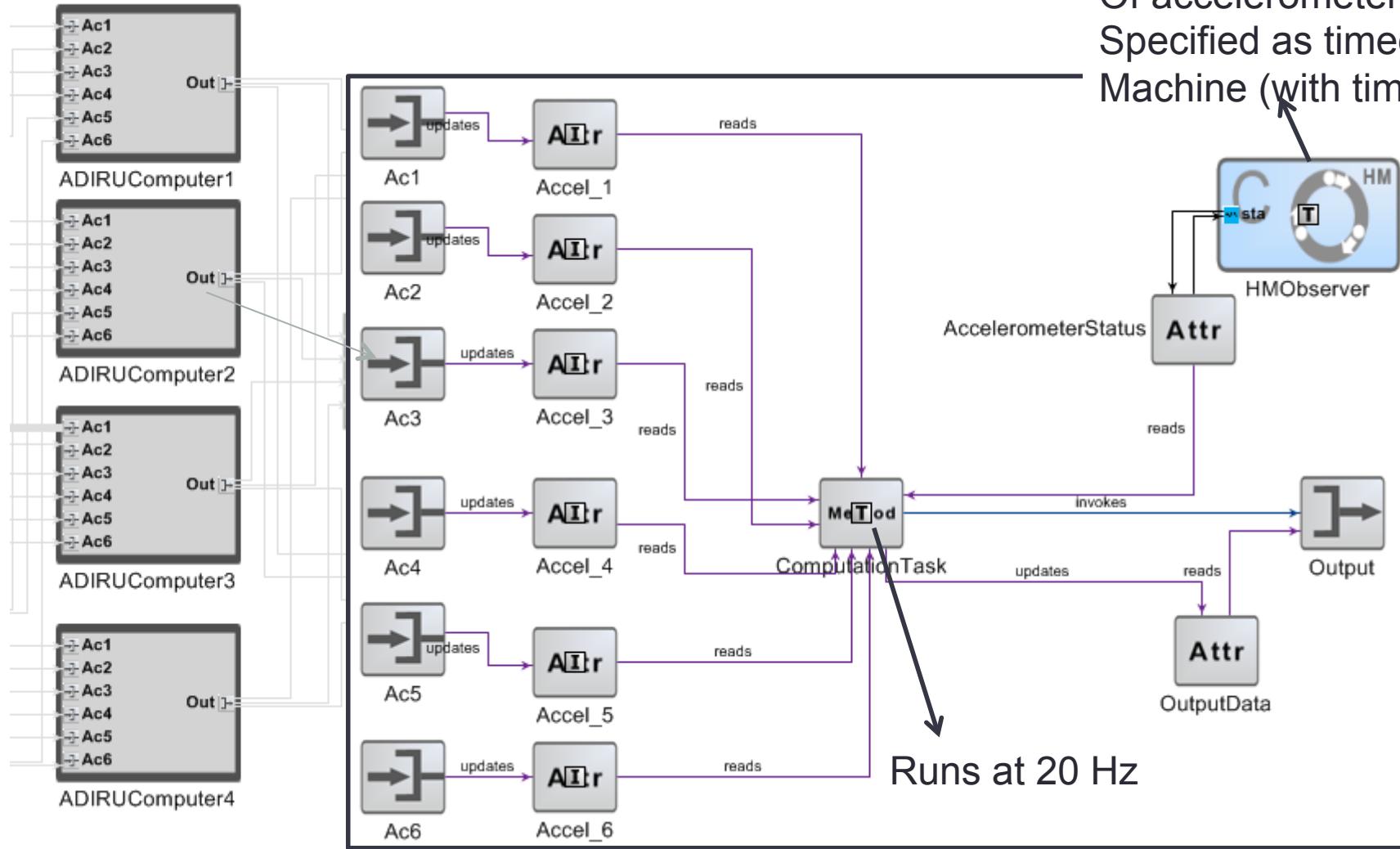
$$\begin{bmatrix} a_1 \\ a_2 \\ a_3 \\ a_4 \\ a_5 \\ a_6 \end{bmatrix} = \begin{bmatrix} -0.3717 & -0.3386 & 0.8644 \\ 0.3717 & -0.8644 & 0.3386 \\ -0.6015 & -0.7971 & -0.0536 \\ -0.9732 & 0.1625 & 0.1625 \\ -0.6015 & -0.0536 & -0.7971 \\ 0.2298 & -0.6882 & -0.6882 \end{bmatrix} \times \begin{bmatrix} a_x \\ a_y \\ a_z \end{bmatrix} + \mathcal{N}$$

# ADIRU Assembly (Accelerometers)

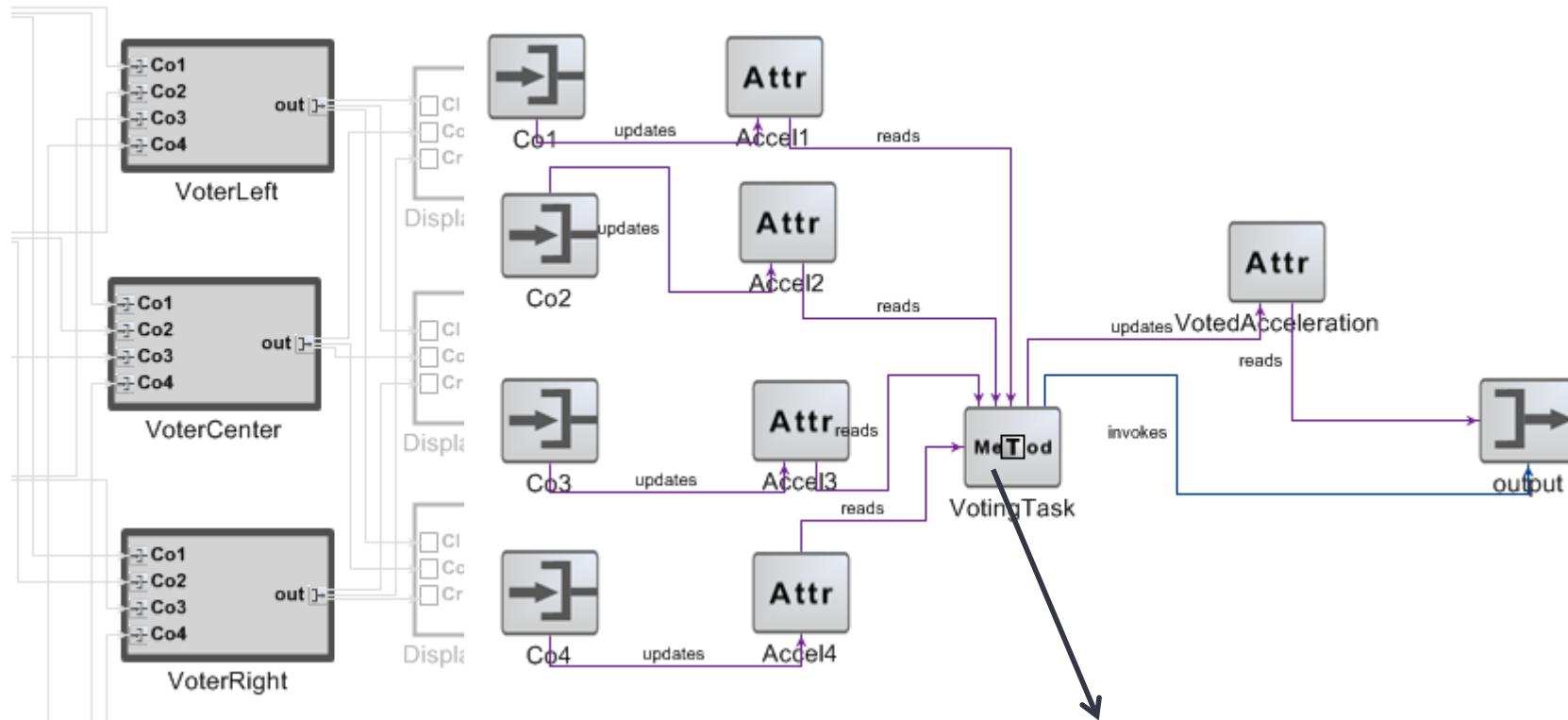


# ADIRU Assembly (Processors)

Observer track the age  
Of accelerometer.  
Specified as timed state  
Machine (with timeout)

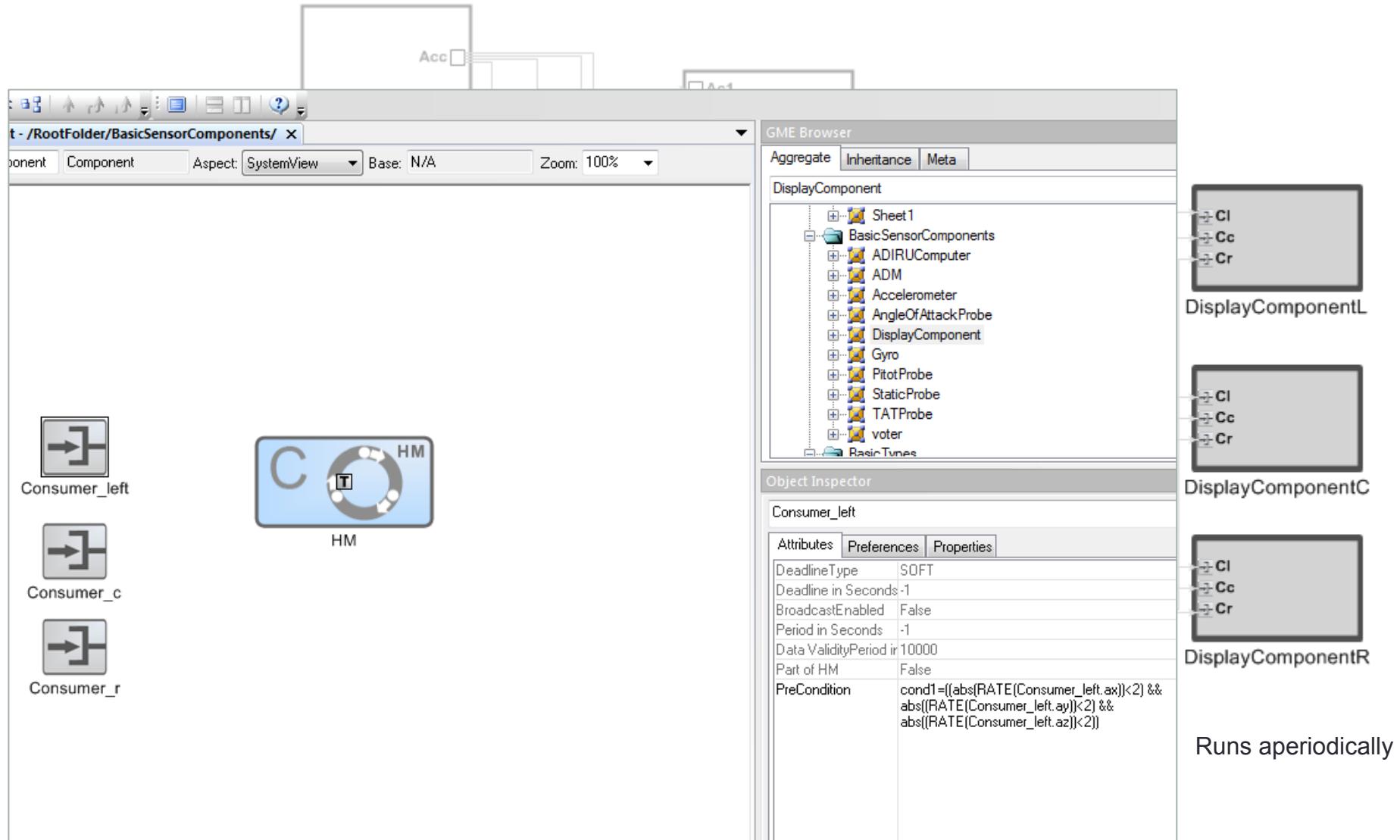


# ADIRU Assembly (Voters)



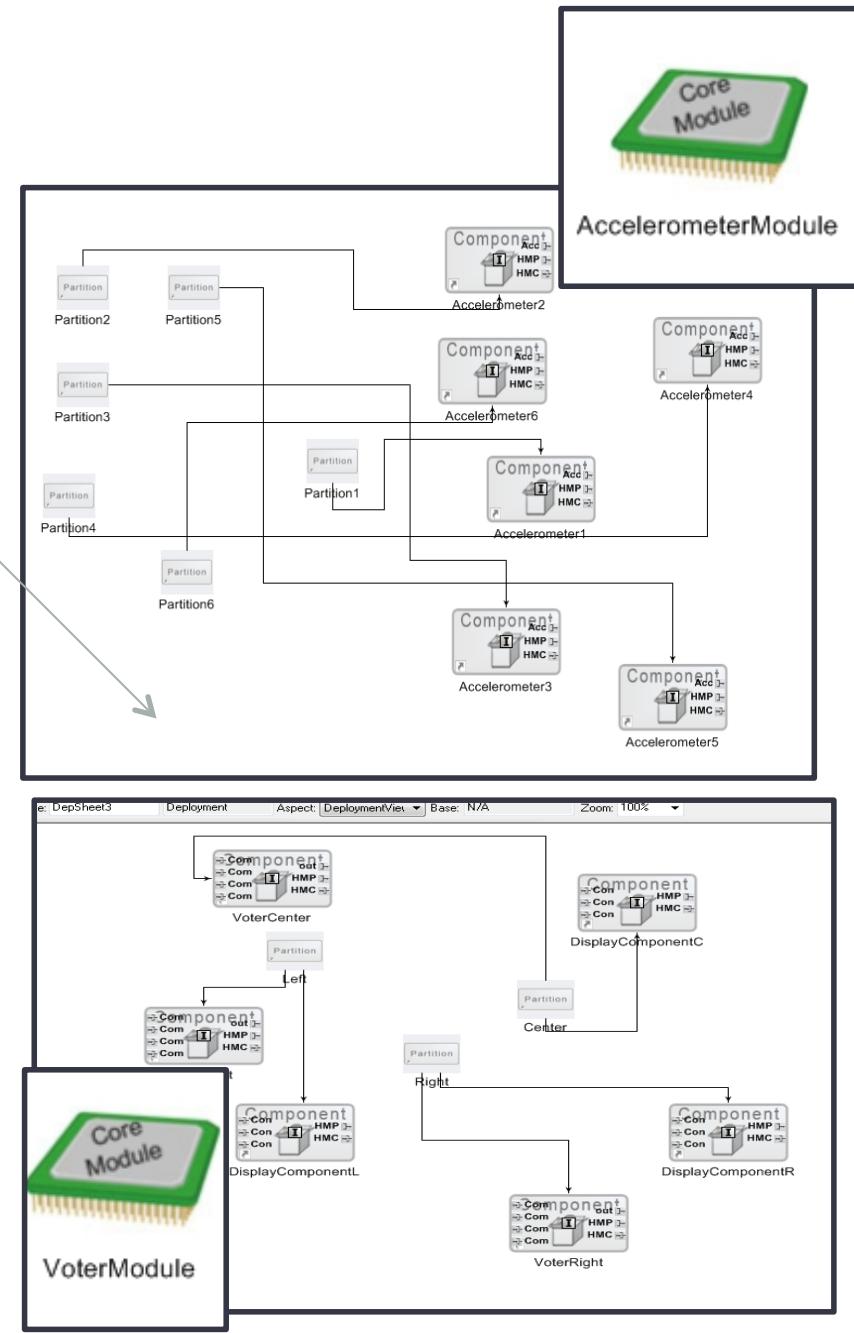
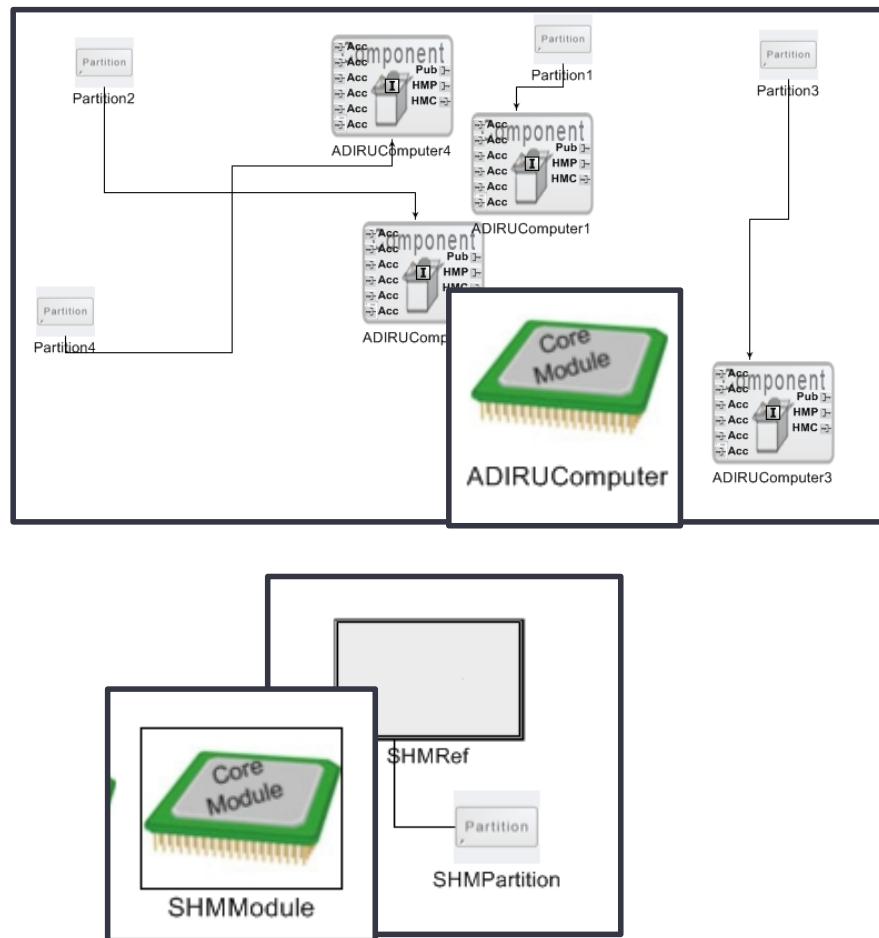
Runs at 20 Hz

# ADIRU Assembly (Display- Mimics PFC)



# Deployment Model

Each Module is a processor running the ARINC Component Runtime Environment



# Execution

## Accelerometer

```
root@durip02:/home/dab
s/Deployments/AccelerometerModule
2684:Partition3|1289939742.610449917|APP|PUBLISH : Accelerometer3_Acceleration
2684:Partition3|1289939742.610486494|APP|Accelerometer3_Impl::APEX_MANAGE_HEALTH Error Received
2684:Partition3|1289939742.610584501|APP|size of error map for process 2, is 1
2684:Partition3|1289939742.610630730|APP|Accelerometer3_Impl::APEX_MANAGE_HEALTH Error Processed
2684:Partition3|1289939742.610642819|APP|Accelerometer3_Impl::APEX_MANAGE_HEALTH start
2684:Partition3|1289939742.610671289|APP|Accelerometer3 :: publishing Acceleration
2684:Partition3|1289939742.610681113|APP|Iter 1,data:3.300000
2684:Partition3|1289939742.610908036|APP| Accelerometer3_APEX_Proc_HMConsumer
2686:Partition4|1289939742.780166269|ERR|Inside run of thread *****
Event state from inside thread is 0
2686:Partition4|1289939742.780188278|ERR|waiting for semaphore Inside run of thread *****
2686:Partition4|1289939742.780235246|APP|Accelerometer4_Impl::APEX_MANAGE_HEALTH start
2686:Partition4|1289939742.780350701|APP|PUBLISH : Accelerometer4_Acceleration
2686:Partition4|1289939742.780362491|APP|Accelerometer4 :: publishing Acceleration
2686:Partition4|1289939742.780370497|APP|Iter 1,data:3.300000
2686:Partition4|1289939742.780588816|APP| Accelerometer4_APEX_Proc_HMConsumer
2688:Partition6|1289939742.950166161|ERR|Inside run of thread *****
Event state from inside thread is 0
2688:Partition6|1289939742.950185488|ERR|waiting for semaphore Inside run of thread *****
2688:Partition6|1289939742.950230636|APP|Accelerometer6_Impl::APEX_MANAGE_HEALTH start
2688:Partition6|1289939742.950344007|APP|PUBLISH : Accelerometer6_Acceleration
2688:Partition6|1289939742.950369112|APP|Accelerometer6_Impl::APEX_MANAGE_HEALTH Error Received
2688:Partition6|1289939742.9504040124|APP|size of error map for process 2, is 1
2688:Partition6|1289939742.950407127|APP|Accelerometer6_Impl::APEX_MANAGE_HEALTH Error Processed
2688:Partition6|1289939742.950479404|APP|Accelerometer6_Impl::APEX_MANAGE_HEALTH start
2688:Partition6|1289939742.950495900|APP|Accelerometer6 :: publishing Acceleration
2688:Partition6|1289939742.950502425|APP|Iter 1,data:3.300000
2688:Partition6|1289939742.950720751|APP| Accelerometer6_APEX_Proc_HMConsumer
```

## ADIRU Processors

```
root@durip09: ~
6163:Part3|1289939742.950888846|APP|ADIRUComputer3_Impl::APEX_MANAGE_HEALTH Error Received
6163:Part3|1289939742.951147546|ERR|waiting for semaphore Inside run of thread *****
*****timer stopped *****
add time invoked for transition 1 time = 1289939744 950900390
6163:Part3|1289939742.951172849|ERR|reset event successful in addtimevalue
Event state from inside addTimeValue is 0
add time invoked for transition 3 time = 1289939744 792925361
6163:Part3|1289939742.951188240|ERR|reset event successful in addtimevalue
Event state from inside addTimeValue is 0
add time invoked for transition 5 time = 1289939744 791734541
6163:Part3|1289939742.951207289|ERR|reset event successful in addtimevalue
Event state from inside addTimeValue is 0
add time invoked for transition 7 time = 1289939744 792323145
6163:Part3|1289939742.951218685|ERR|reset event successful in addtimevalue
Event state from inside addTimeValue is 0
add time invoked for transition 9 time = 1289939744 793527434
6163:Part3|1289939742.951234566|ERR|reset event successful in addtimevalue
Event state from inside addTimeValue is 0
6163:Part3|1289939742.951254867|ERR|*****startTimer Called
6163:Part3|1289939742.951268307|ERR|Running Timer
I am here in the run *****
*****timer started *****
6163:Part3|1289939742.951294267|APP|size of error map for process 19, is 1
6163:Part3|1289939742.951321255|APP|ADIRUComputer3_Impl::APEX_MANAGE_HEALTH Error Processed
6163:Part3|1289939742.951330592|APP|ADIRUComputer3_Impl::APEX_MANAGE_HEALTH start
6163:Part3|1289939742.951343945|APP|ADIRUComputer3 :: consuming Ac6
6163:Part3|1289939742.951354760|APP|Accel_6: 0000
6163:Part3|1289939742.951360902|APP|Accel_6: 0000
6163:Part3|1289939742.951397696|APP| ADIRUComputer3_APEX_Proc_Ac6
```

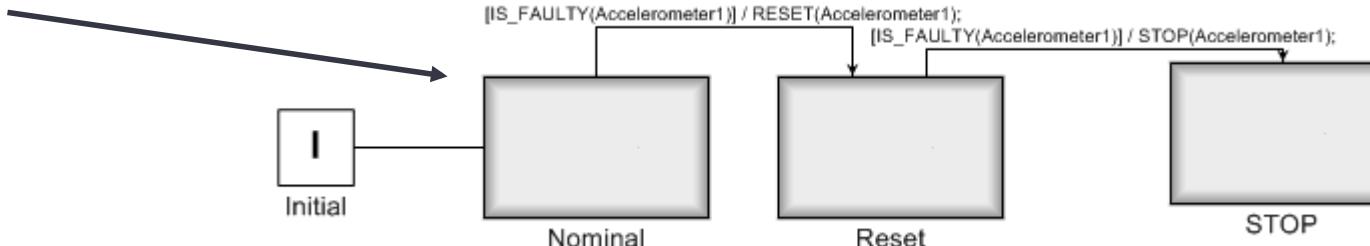
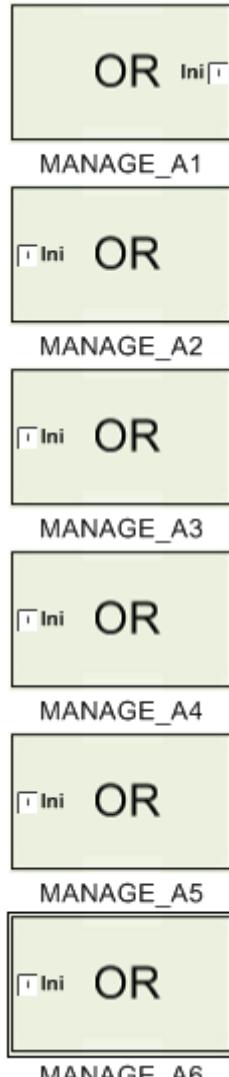
```
root@durip09: ~/AbhishekSharedFolder/noutput4a/gen/Deployment/Deployments/SHMModule
7437:SHMPartition|1289939740.275260294|APP|DiagnosisEngine_Impl:: fm_FM_VoterLeft_USER_CODE_PROBLEM
comp VoterLeft
7437:SHMPartition|1289939740.275282381|APP|DiagnosisEngine_Impl:: fm_FM_VoterRight_LOCK_PROBLEM
comp VoterRight
7437:SHMPartition|1289939740.275294268|APP|DiagnosisEngine_Impl:: fm_FM_VoterRight_USER_CODE_PROBLEM
comp VoterRight
7437:SHMPartition|1289939740.275309238|APP|DiagnosisEngine_Impl::initDiagnoser - finished
7437:SHMPartition|1289939740.275320135|APP|DiagnosisEngine - Created
7437:SHMPartition|1289939740.275460393|APP|DiagnosisEngine Component - finished home creation and activation
7437:SHMPartition|1289939740.275487640|APP|DiagnosisEngine Component - created
7437:SHMPartition|1289939740.275520561|APP|SHMEngine - Activated
7437:SHMPartition|1289939740.275545239|APP|AssemblyHM - Activated
7437:SHMPartition|1289939740.275560542|APP|DiagnosisEngine - Activated
7437:SHMPartition|1289939741.278793168|APP|SHMEngine - Activated
7437:SHMPartition|1289939741.278823261|APP|AssemblyHM - Activated
7437:SHMPartition|1289939741.278850020|APP|DiagnosisEngine - Activated
7437:SHMPartition|1289939741.281072120|APP|Setting Partition State
7437:SHMPartition|1289939742.290226177|ERR|Inside run of thread *****
Event state from inside thread is 0
7437:SHMPartition|1289939742.290255277|ERR|waiting for semaphore Inside run of thread *****
7437:SHMPartition|1289939742.290488690|APP| SHMEngine_APEX_Proc_Consumer
7437:SHMPartition|1289939742.290592525|APP|PUBLISH : AssemblyHM_AlarmPublisher
7437:SHMPartition|1289939742.290618958|HME|HME Not available for process id 21
7437:SHMPartition|1289939742.290629248|APP|AssemblyHM :: publishing AlarmPublisher
7437:SHMPartition|1289939742.290640884|APP|AssemblyHM :: event Buffer empty
7437:SHMPartition|1289939742.290737853|APP| AssemblyHM_APEX_Proc_HMConsumer
7437:SHMPartition|1289939742.290826332|APP| proc_AlarmConsumer
```

## SHM

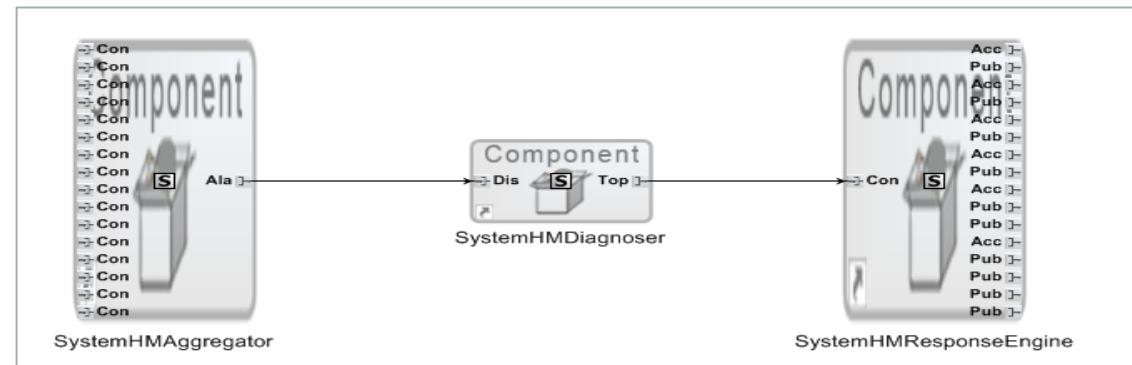
## VOTERS + DISPLAY

```
root@durip09: ~
8245:Center|1289939742.620605050|ERR|waiting for semaphore Inside run of thread *****
8245:Center|1289939742.620749228|APP|DisplayComponentC_Impl::APEX_MANAGE_HEALTH start
8245:Center|1289939742.620947290|APP| VoterCenter_APEX_Proc_C01
8245:Center|1289939742.621039519|APP| VoterCenter_APEX_Proc_C02
8245:Center|1289939742.621112732|APP| VoterCenter_APEX_Proc_C03
8245:Center|1289939742.621195766|APP| VoterCenter_APEX_Proc_C04
8245:Center|1289939742.621268694|APP| VoterCenter_APEX_Proc_HMConsumer
8245:Center|1289939742.621347821|APP| DisplayComponentC_APEX_Proc_C1
8245:Center|1289939742.621390953|APP| DisplayComponentC_APEX_Proc_Cc
8245:Center|1289939742.621436150|APP| DisplayComponentC_APEX_Proc_Cr
8245:Center|1289939742.621478667|APP| DisplayComponentC_APEX_Proc_HMConsumer
8245:Center|1289939742.621599151|APP|METHOD : COMP_VoterCenter_VotingTask
8245:Center|1289939742.621625558|HME|HME Not available for process id 32
8245:Center|1289939742.621636508|APP|ComputationTask: BodyAccelerationValue a1: ax=0.000000,ay=0.000000,az=0.000000
8245:Center|1289939742.621648062|APP|ComputationTask: BodyAccelerationValue a2: ax=0.000000,ay=0.000000,az=0.000000
8245:Center|1289939742.621658636|APP|ComputationTask: BodyAccelerationValue a3: ax=0.000000,ay=0.000000,az=0.000000
8245:Center|1289939742.621673021|APP|ComputationTask: BodyAccelerationValue a4: ax=0.000000,ay=0.000000,az=0.000000
8245:Center|1289939742.62168808|APP|Stale Data ComputationTask: BodyAccelerationValue a1
8245:Center|1289939742.621697770|APP|Stale Data ComputationTask: BodyAccelerationValue a2
8245:Center|1289939742.621705821|APP|Stale Data ComputationTask: BodyAccelerationValue a3
8245:Center|1289939742.621719797|APP|Stale Data ComputationTask: BodyAccelerationValue a4
```

# System Health Manager



other machines have similar specification



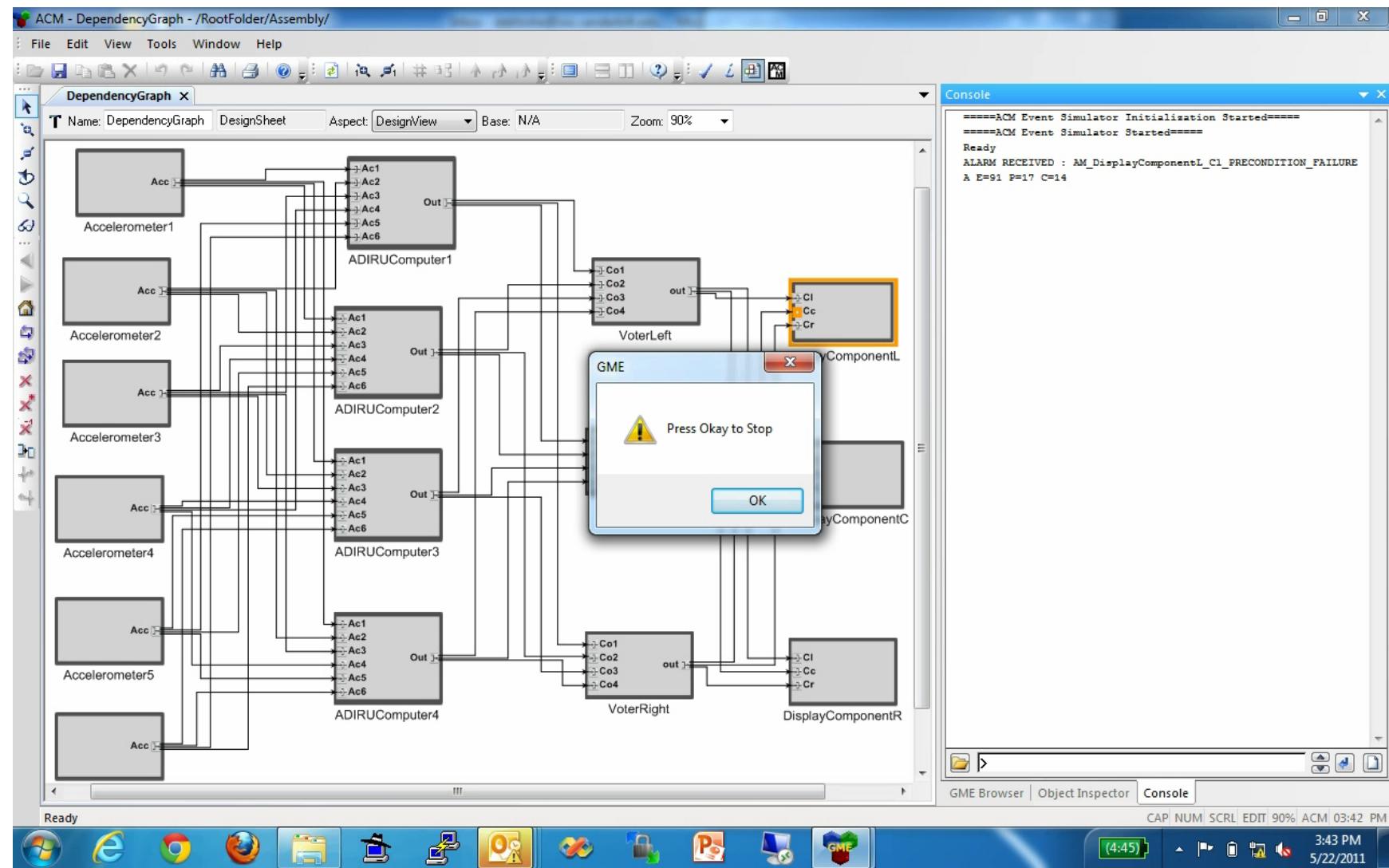
These components are auto generated

The hypothesis generated by the diagnoser is translated to Component(s) that is most likely faulty. This list is fed to Response Engine, which triggers the mitigation state machine

# Demonstration

- Fault Scenario
- Accelerometer 5 has initial fault
- It is started which causes an alarm
- Then Accelerometer 6 develops fault
- Successful mitigation
  - Identifying the faulty components
  - Stopping the fault components
  - Processors can still function with four accelerometers.

# Demonstration: Faulty Scenario (Movie)



# Conclusion

- Software health management is feasible and affordable using a model-based, component-oriented approach.
- We showed the results of an experiment that simulated partial abilities of a Boeing 777 ADIRU.
- Component framework is essential
  - Managed component interactions
  - Monitored interfaces
- Architecture and component models are needed for deriving fault models and to determine correct mitigation actions
- Fault diagnostics across components is relevant to isolate root causes
- Mitigation can be reactive (with pre-determined reactions) or deliberative (with dynamically computed reactions)
- Deliberative mitigation is the subject of ongoing research.

