A Software System for Real-Time Fire Monitoring (RTFM)

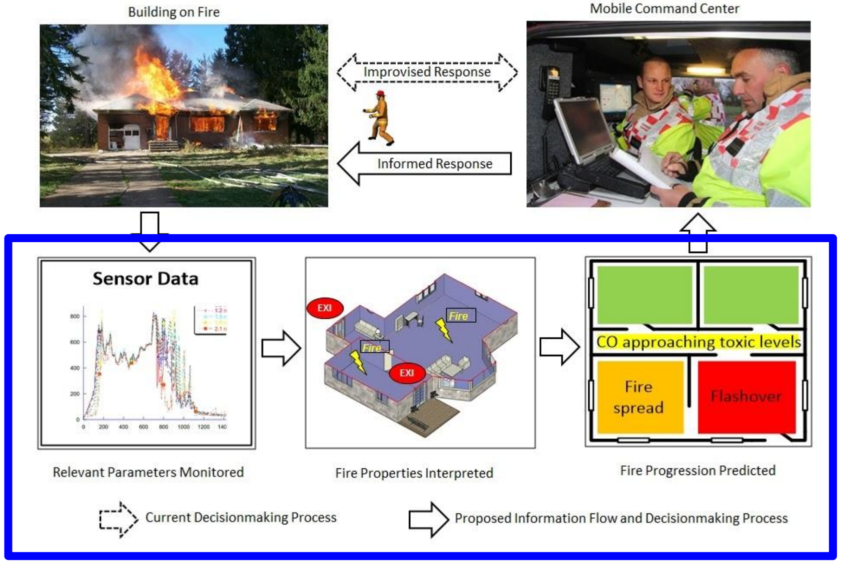
- **PI: Professor Ann Jeffers**
- **Ph.D. Student: Paul A. Beata**
- **University of Michigan in Ann Arbor, MI**
- **Dept. of Civil and Environmental Engineering**

Project Goal: ***To develop a flexible computing infrastructure for integrating real-time fire data into simulation and visualization software***

Project Components:

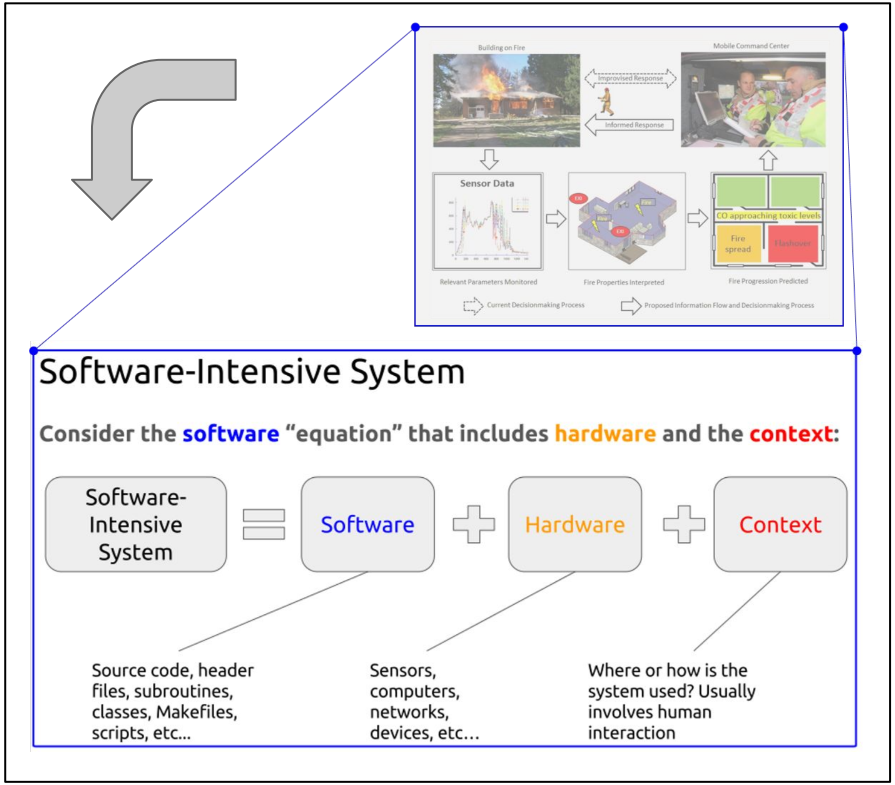
1. Simulate fire data using CFD fire models
2. Use message-passing and sub-models for real-time fire monitoring
3. Distributed computation and visualization

Proposed **Distributed Deployment of system for improved fire-fighting and decision making:**



Building on Fire
Mobile Command Center
Improvised Response
Informed Response
Sensor Data
EXI
Fire
Fire
EXI
CO approaching toxic levels
Fire spread
Flashover
Relevant Parameters Monitored
Fire Properties Interpreted
Fire Progression Predicted
Current Decisionmaking Process
Proposed Information Flow and Decisionmaking Process

A Software System for Real-Time Fire Monitoring (RTFM)


Building on Fire
Mobile Command Center
Improvised Response
Informed Response
Sensor Data
CO approaching toxic levels
Fire spread
Flashover
Relevant Parameters Monitored
Fire Properties Interpreted
Fire Progression Predicted
Current Decisionmaking Process
Proposed Information Flow and Decisionmaking Process

Software-Intensive System

Consider the software "equation" that includes hardware and the context:

Software-Intensive System = Software + Hardware + Context

Source code, header files, subroutines, classes, Makefiles, scripts, etc...

Sensors, computers, networks, devices, etc...

Where or how is the system used? Usually involves human interaction

This project focuses on the **software engineering** process for providing the RTFM.

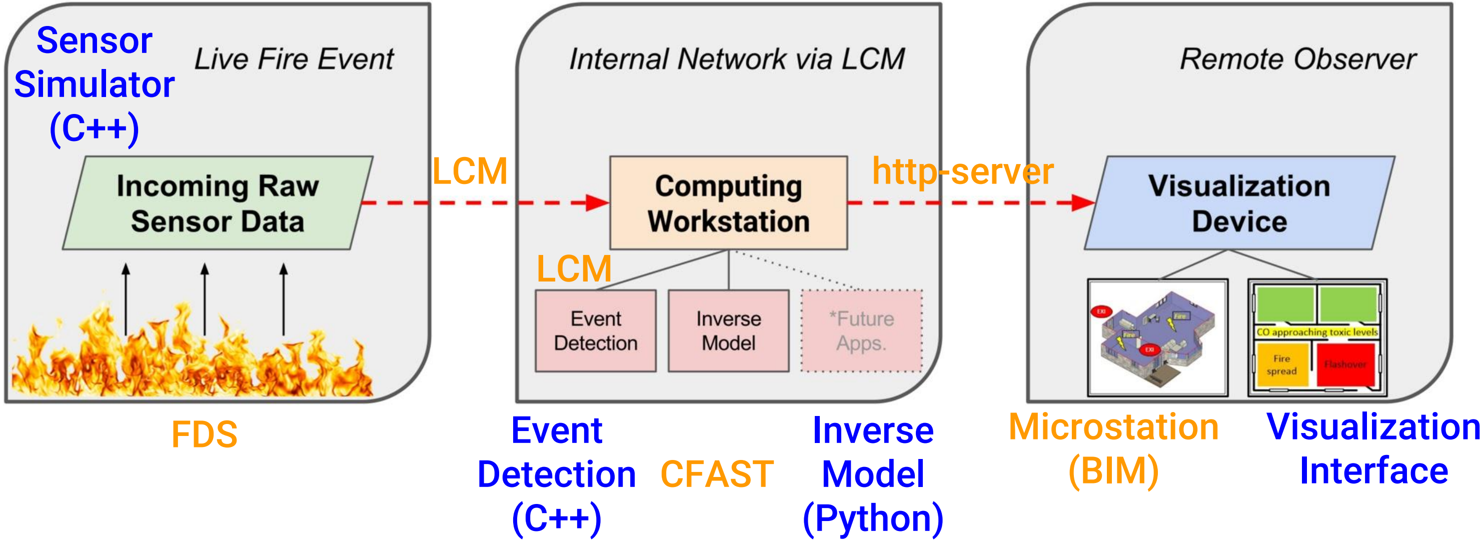
The image shows the proposed framework represented as a ***software-intensive system***.

For this project, use and develop new tools:

1. Use FDS from NIST to generate fire data
2. Use CFAST from NIST to compute heat rate
3. Use LCM for message-passing to/from models
4. Use http-server for distributed deployment
5. Develop sensor simulator to push new data: C++
6. Develop inverse fire model (pred. heat rate): Python
7. Develop fire event-detection model: C++
8. Develop main event loops for handling data: Python
9. Develop visualization tool for Bentley Microstation

A Software System for Real-Time Fire Monitoring (RTFM)

To combine these tools (CFAST, LCM, etc.), we propose the following components: [blue in development]



Sensor Simulator (C++)

Live Fire Event

Incoming Raw Sensor Data

FDS

LCM

Internal Network via LCM

Computing Workstation

LCM

Event Detection

Inverse Model

*Future Apps.

http-server

Remote Observer

Visualization Device

Event Detection (C++)

CFAST

Inverse Model (Python)

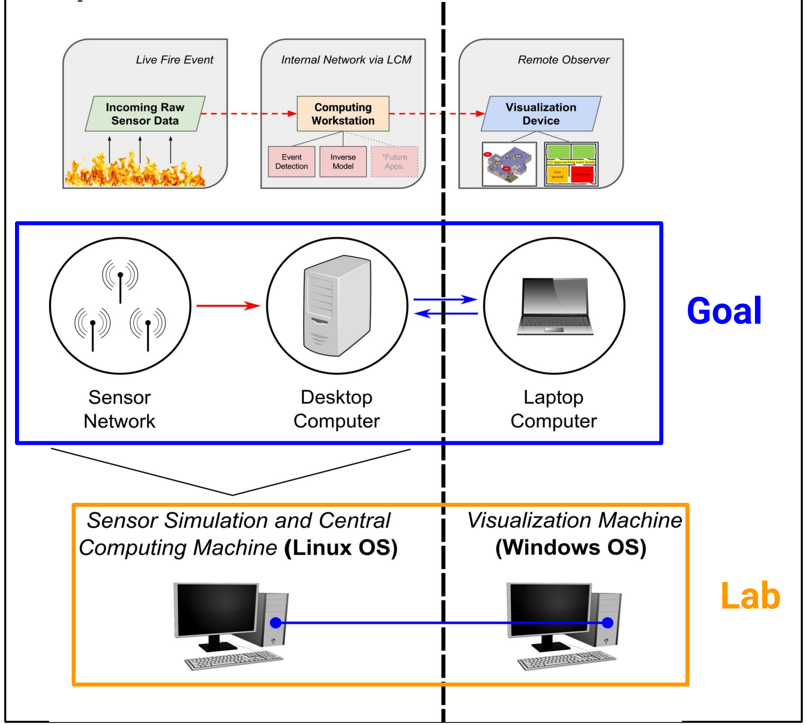
Microstation (BIM)

Visualization Interface

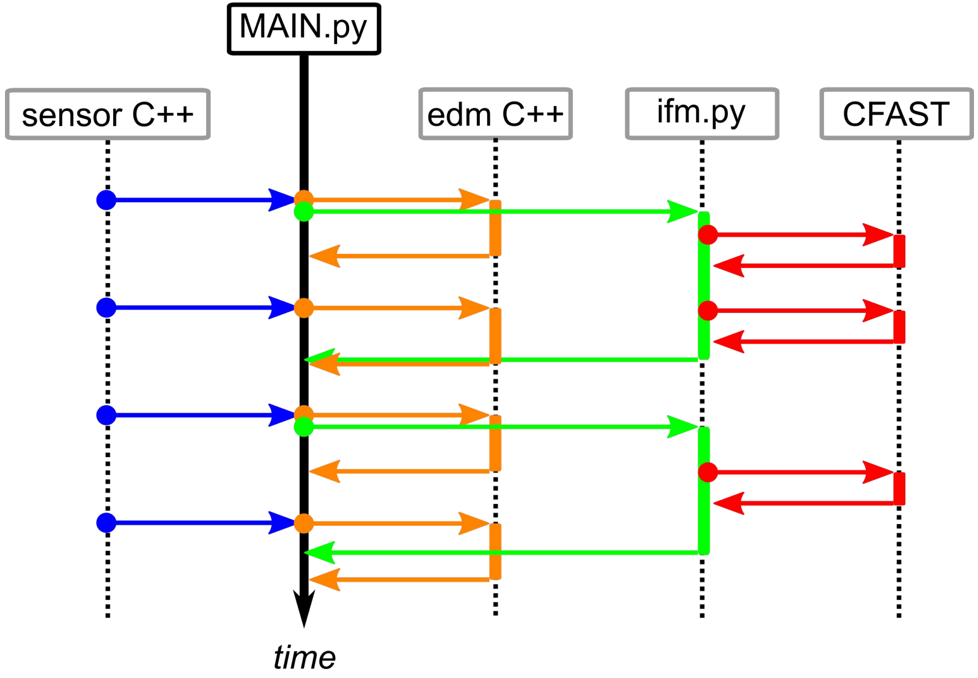
Main Program (Python): Facilitates data transfer via LCM and handles the main event loop for simulating the real-time fire scenario.

A Software System for Real-Time Fire Monitoring (RTFM)

Implementation in Controlled Environment:



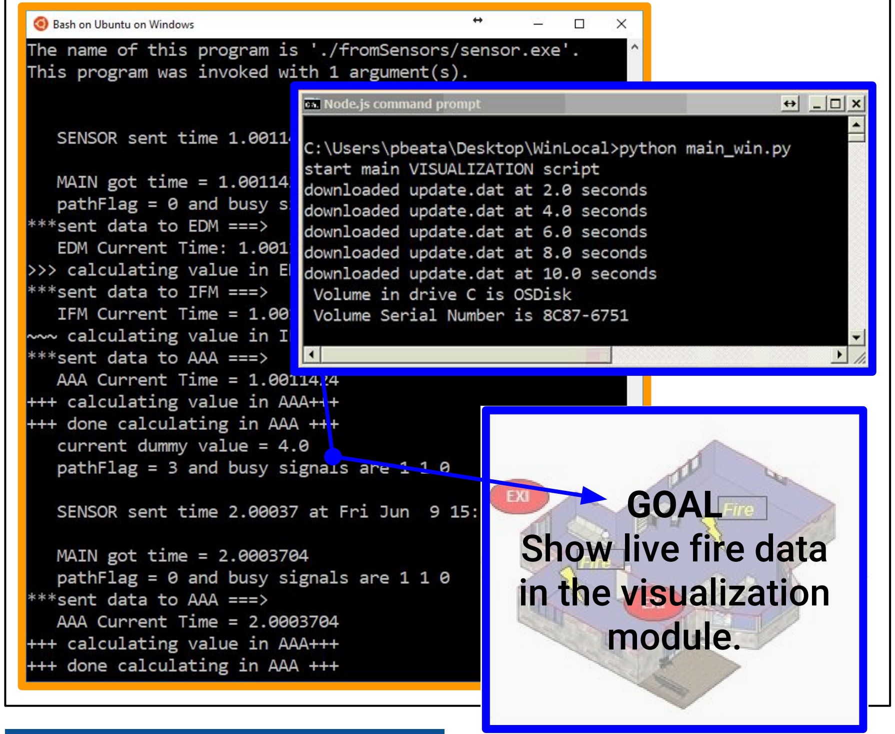
Subprocess Timeline with Python (Compute Machine):



University of Michigan

A Software System for Real-Time Fire Monitoring (RTFM)

Sample Output (*incomplete system*):



```
The name of this program is './fromSensors/sensor.exe'.
This program was invoked with 1 argument(s).


   SENSOR sent time 1.0011

   MAIN got time = 1.00114
   pathFlag = 0 and busy s
***sent data to EDM ===>
   EDM Current Time: 1.001
>>> calculating value in E
***sent data to IFM ===>
   IFM Current Time = 1.00
~~~ calculating value in I
***sent data to AAA ===>
   AAA Current Time = 1.0011424
+++ calculating value in AAA+++
+++ done calculating in AAA +++
   current dummy value = 4.0
   pathFlag = 3 and busy signals are 1 1 0

   SENSOR sent time 2.00037 at Fri Jun  9 15:

   MAIN got time = 2.0003704
   pathFlag = 0 and busy signals are 1 1 0
***sent data to AAA ===>
   AAA Current Time = 2.0003704
+++ calculating value in AAA+++
+++ done calculating in AAA +++
```

```

C:\Users\pbeata\Desktop\WinLocal>python main_win.py
start main VISUALIZATION script
downloaded update.dat at 2.0 seconds
downloaded update.dat at 4.0 seconds
downloaded update.dat at 6.0 seconds
downloaded update.dat at 8.0 seconds
downloaded update.dat at 10.0 seconds
 Volume in drive C is OSDisk
 Volume Serial Number is 8C87-6751
```

GOAL
Show live fire data in the visualization module.

Future Work:

- A high-level view of the proposed infrastructure was presented here, but the visualization component is incomplete

- We aim to package the mature computing workstation components into a single repository and use automatic build tools

- The final product will be a complete fire-monitoring simulation which is user-friendly and appealing to non-scientists

University of Michigan