Space Telecommunications Radio System

STRS Cognitive Radio

SDR-WInnComm 2013

Presented by:
Janette C Briones
Louis M Handler
NASA Glenn Research Center

NASA Space Communications Needs

May vary over a wide range of operating conditions:

1. Autonomous operation (generally no operator)
2. Large distances and time delays between radios
3. Large relative velocities between radios
4. Large temperature changes
5. Intermediate environmental factors (atmospheric, solar, and cosmic effects)
6. Limited resources (mainly power)
7. Higher frequency waveforms
8. Unreplaceable deployed radios

Cognitive Radio Architecture

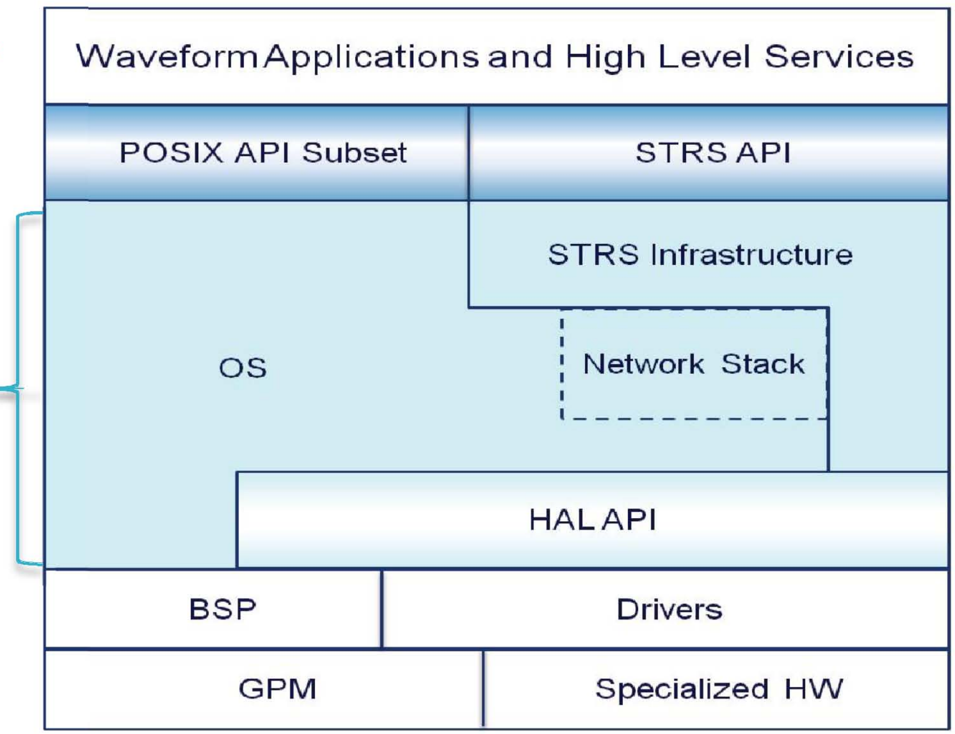
- An SDR provides the most capability for integrating autonomic decision-making ability and changing functionality as needed.

- Standardized architecture provides consistency and allows leveraging of past success.

- An STRS radio is optimized for space and therefore STRS will be considered in the following slides.

STRS Layer Cake

- Software
- Interface
- Operating Environment
- Firmware



Cognitive Engine Learning Information

1. Link State
   - Current channel bandwidth, observed bit error rate (BER), round trip time, transmit power, and the encoding overhead of the current modulation scheme.

2. Application State
   - Carrier frequency, modulation scheme, symbol rate, etc.

Cognitive Engine Learning Information

3. Spatial State
   - Relative physical locations of other nodes in a network may be available.

4. Environment State
   - Current local environment of the node, such as available electrical power, temperature and other health information.

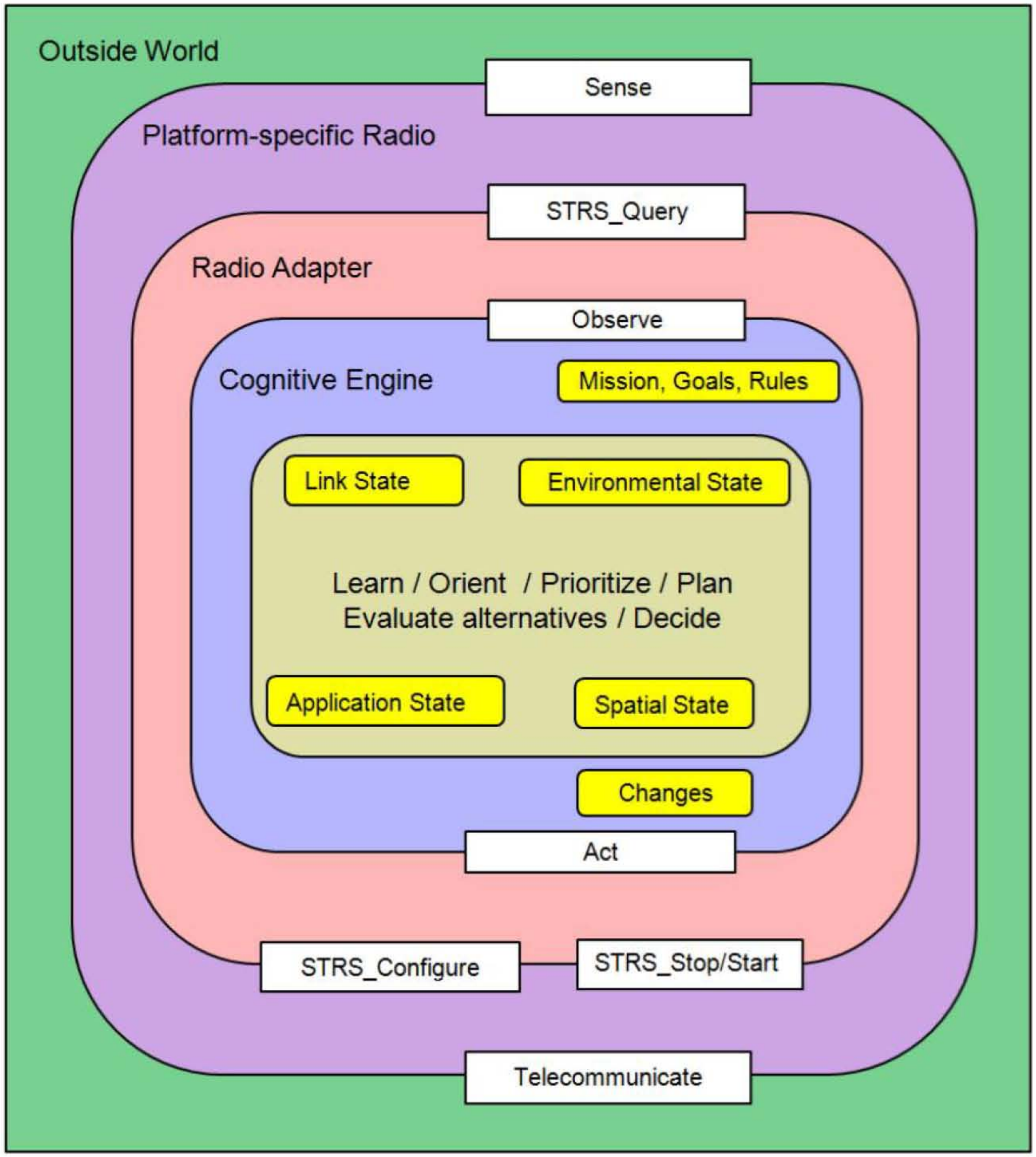
Cognitive Engine Learning Information

5. Radio Platform Information
   - Configuration parameter set of the radio platform to carry out required waveform and link operations.

6. Mission Information
   - What has to be done, with what kind of radios, where, and when.

STRS Cognitive Layers

(Egg Model)



Cognitive Radio Observer

- Passes the data to the Cognitive Engine in a format that it can use for further processing.
- May obtain data according to events (e.g. time), or as needed.
- Uses the SDR's API to obtain the data from the SDR to support learning and decision-making.
- May be split into SDR adapters and the cognitive engine's observers depending on how the cognitive engine needs to obtain data.

STRS Cognitive Radio Observer

- Simple observations would be obtained using STRS_Query.
- More complicated observations might be obtained using STRS_RunTest.
- Even more complicated observations might be obtained using STRS applications and services that monitor some beacon or other external signal.

Cognitive Radio Actor

- Obtains actions from the Cognitive Engine.
- Uses the SDR's API to perform those actions.
- May be split into an SDR adapter and the cognitive engine's actors according to functions it needs to perform.

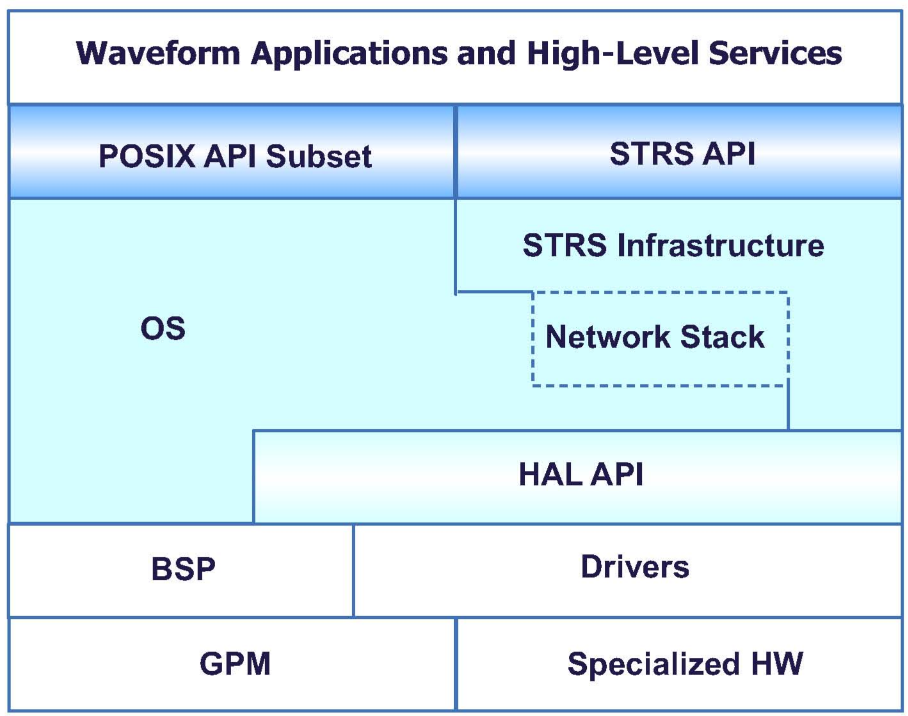
STRS Cognitive Radio Actor

- Simple actions would use STRS_Configure, STRS_Start, and STRS_Stop.

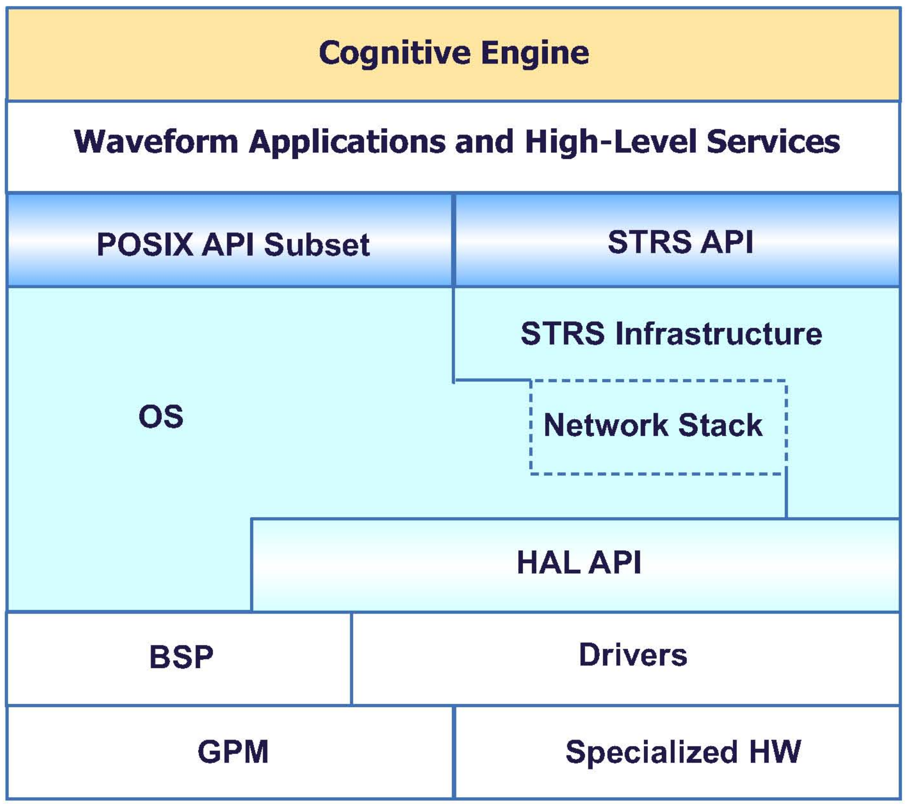
- More complicated actions would use STRS_AbortApp and STRS_InstantiateApp.

- Even more complicated actions would download and/or create new applications to implement necessary functionality.

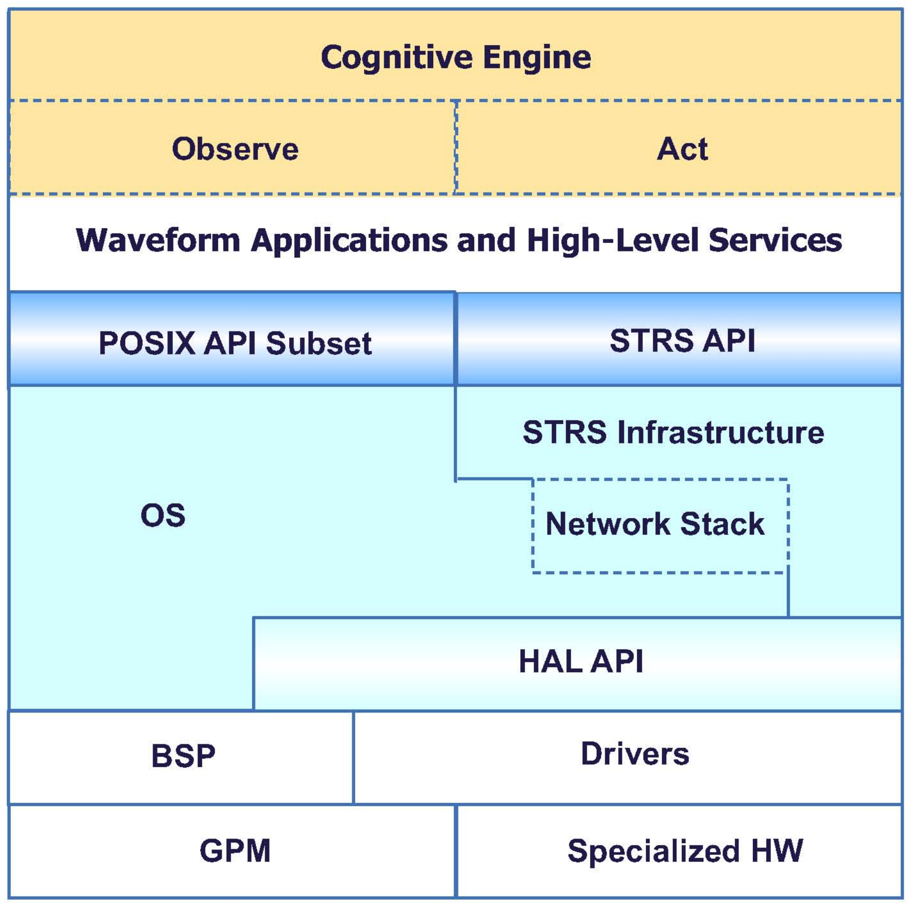
STRS Layer Cake



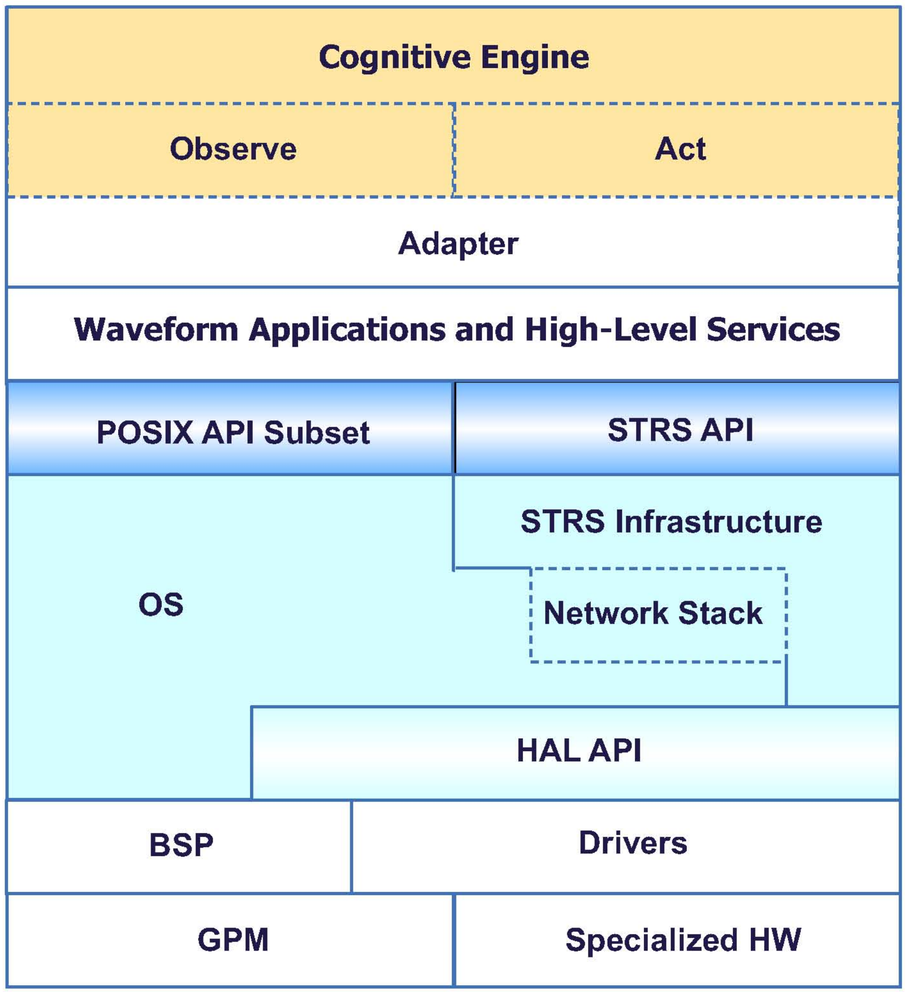
STRS Layer Cake



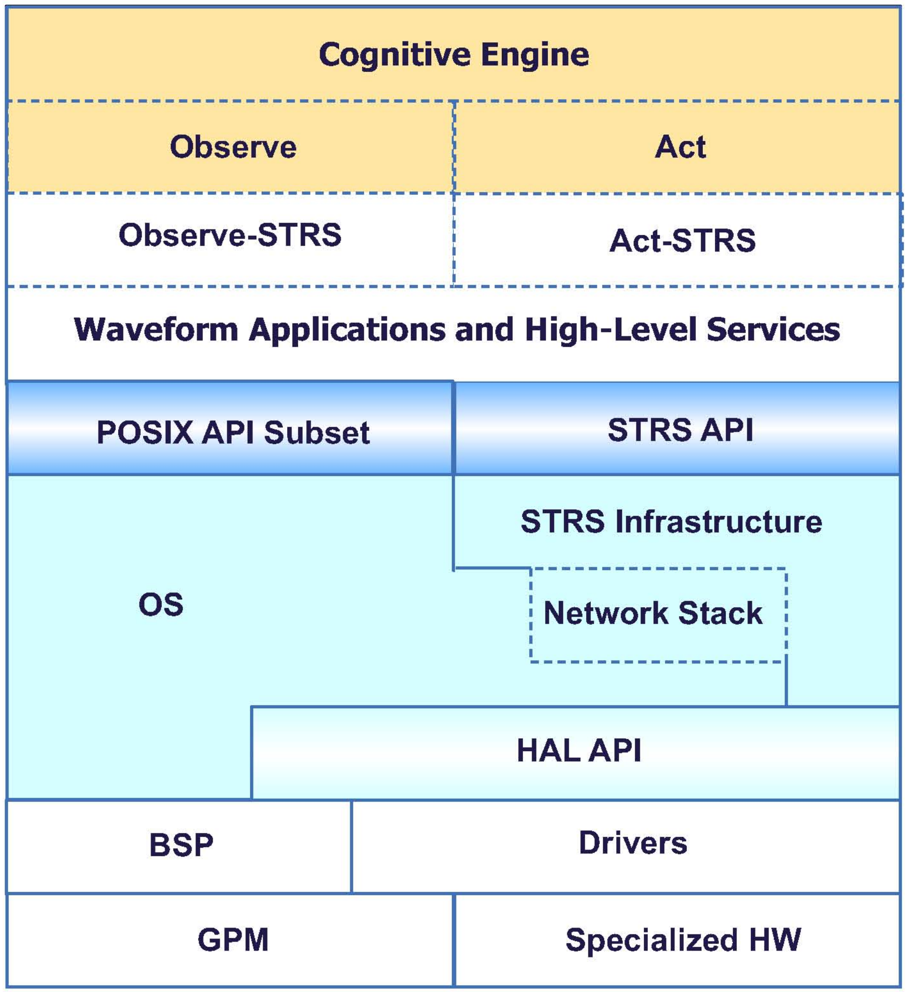
STRS Layer Cake



STRS Layer Cake



STRS Layer Cake



Cognition Cycle

- Incorporating a Cognitive Engine into a radio leads to the development of a cognition cycle that loops through:

Plan

LEARN

DECIDE

OBSERVE

Orient

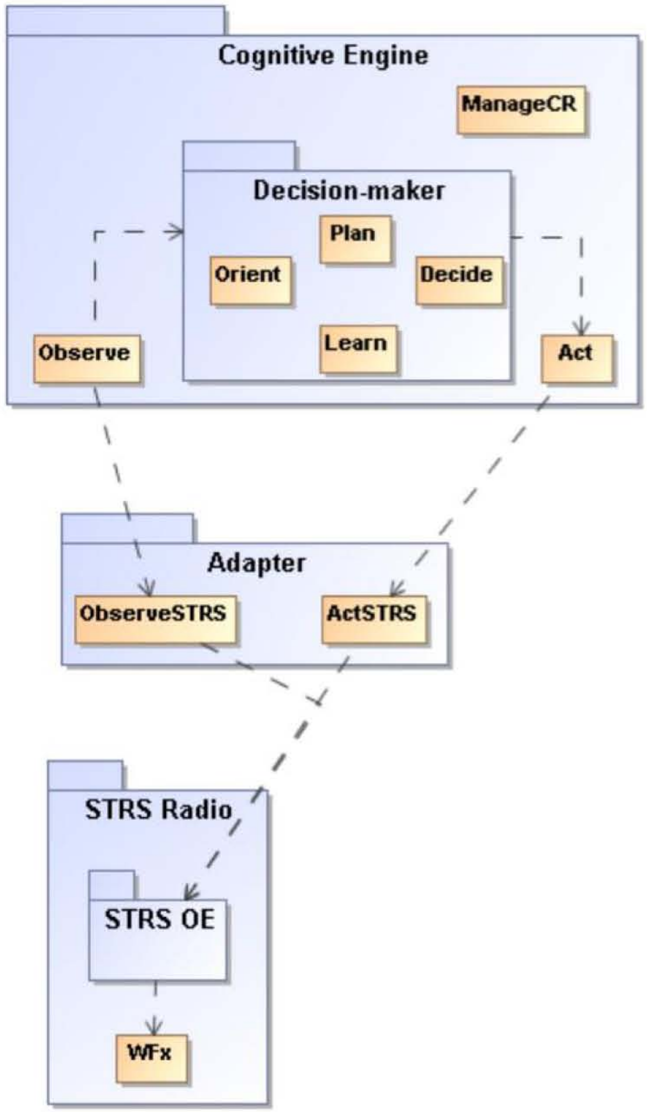
ACT

Cognitive Engine

- The Cognitive Engine will evolve as needed when considering the complexity of the SDR and the decision-making process.

- Additional pieces might include:
  - Neural network
  - Data mining
  - Scheduler
  - Cognitive engine manager shown in the next slide as ManageCR

Cognitive Layers in UML



Configuration File Data

- Configure what data is observed and what STRS commands will obtain the data.
- Configure what actions are allowed and what STRS commands will accomplish the desired action.
- Configure pre-defined rules for the mission, prioritized goals, and equations.

Implementing a Cognitive Radio

Start with a particular STRS radio implementation:

- Understand the capabilities of the radio, the mission requirements, the allowed actions, and supporting data.
- Design how the data will be observed, decisions made, and actions performed.
- The cognitive engine may be integrated into the radio as a separate process.

Conclusion

- Will build upon the infrastructure developed by STRS SDR technology
  - Adds cognitive capabilities with interfaces between the cognitive engine and the STRS radio.
  - Allows technology improvements to optimize radio performance and autonomous decision-making based on feedback.

- Will create new capabilities for new science opportunities.
  - Cognitive engine's learning can influence what we do and further our knowledge about telecommunications and networking under extreme conditions.
  - As we understand more about the cognitive radio's successes and failures, we can add additional observed variables and functionality to improve performance.

- Can lower the operational costs and improve performance of NASA's future radios.

Questions