

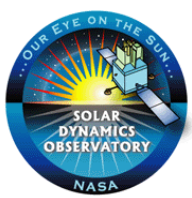


Processing Module Development

Rasmus Munk Larsen, Stanford University

rmunk@quake.stanford.edu

650-725-5485



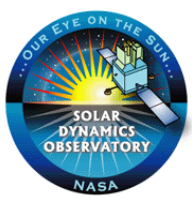
Overview

HMI

Stanford University

LOCKHEED MARTIN
Advanced Technology Center

- **New challenges compared to MDI**
- **Module status and MDI heritage**
- **Module structure and development strategy**
- **Community contributions and collaboration**



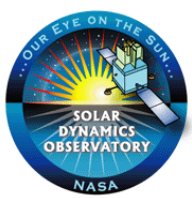
HMI Data Processing Pipeline

HMI

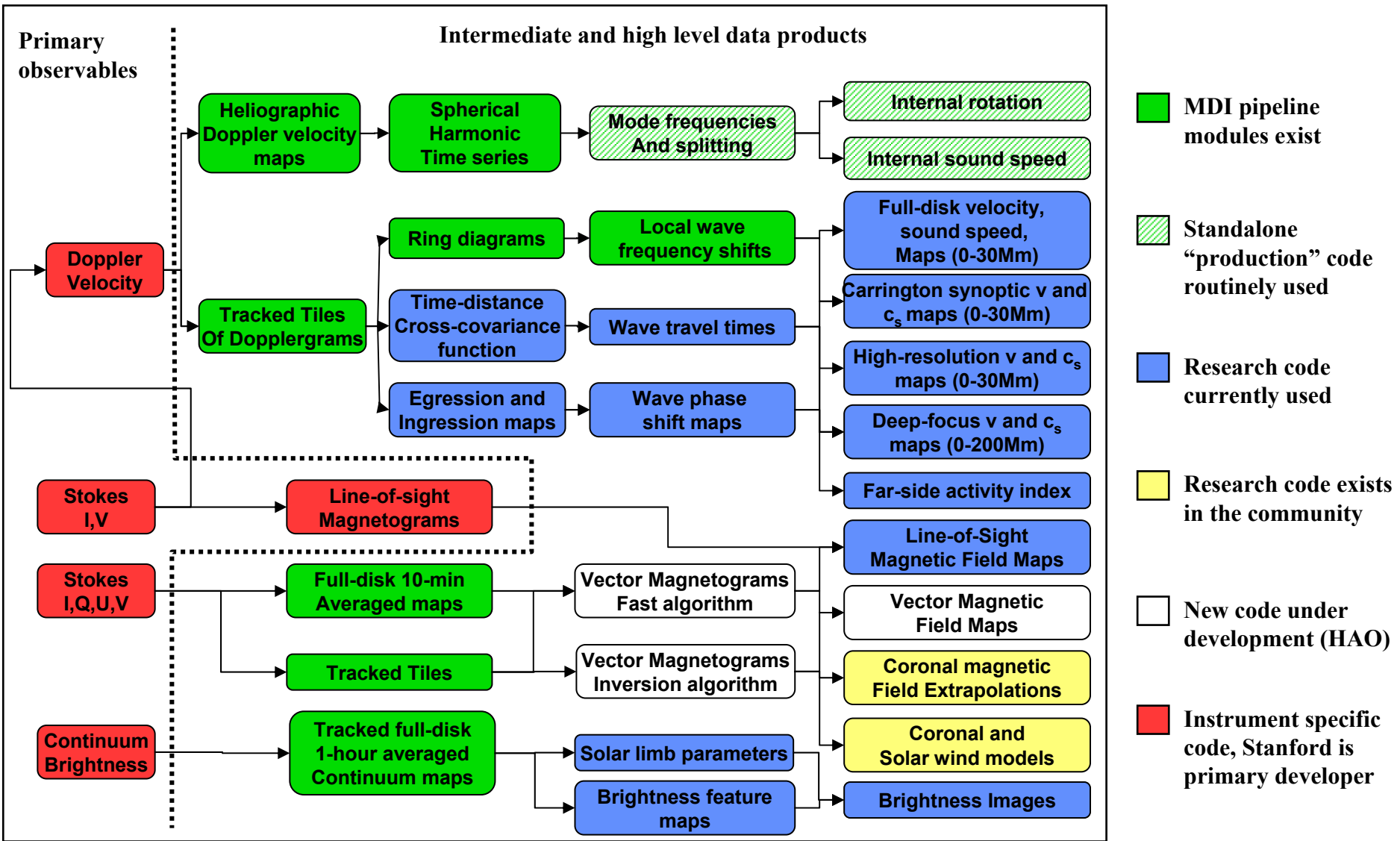
Stanford University

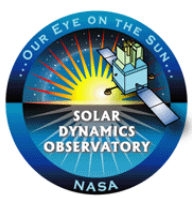
LOCKHEED MARTIN
Advanced Technology Center

- **New challenges compared to MDI:**
 - Real-time and reliability requirements on ground based observable calculations
 - Level 0-1 of pipeline (telemetry capture, observable calculations) must support instrument ground testing
 - Real-time requirements on high level data products for space weather etc. demands dual pipeline paths:
 - › Fast algorithms for preliminary (“quick look”) data products
 - › Slower, more accurate process for definitive calibrated data products
 - Pipeline mode generation of high-level data products
 - Automatic on-demand generation of data products, e.g., triggered by VSO queries
 - › Requires improved traceability, in particular when using evolving research codes
 - Vastly increased data volume
 - New computationally intensive data analyses, such as time-distance analysis, farside imaging, vector magnetogram inversion
 - Vector magnetic data products and processing
- **Ameliorating circumstances**
 - Large body of knowledge and software from MDI, GONG & other projects
 - Moore’s law (Disk density, networking bandwidth, computing power)
 - Maturing computing infrastructures (Web technologies, Grids, software tools)

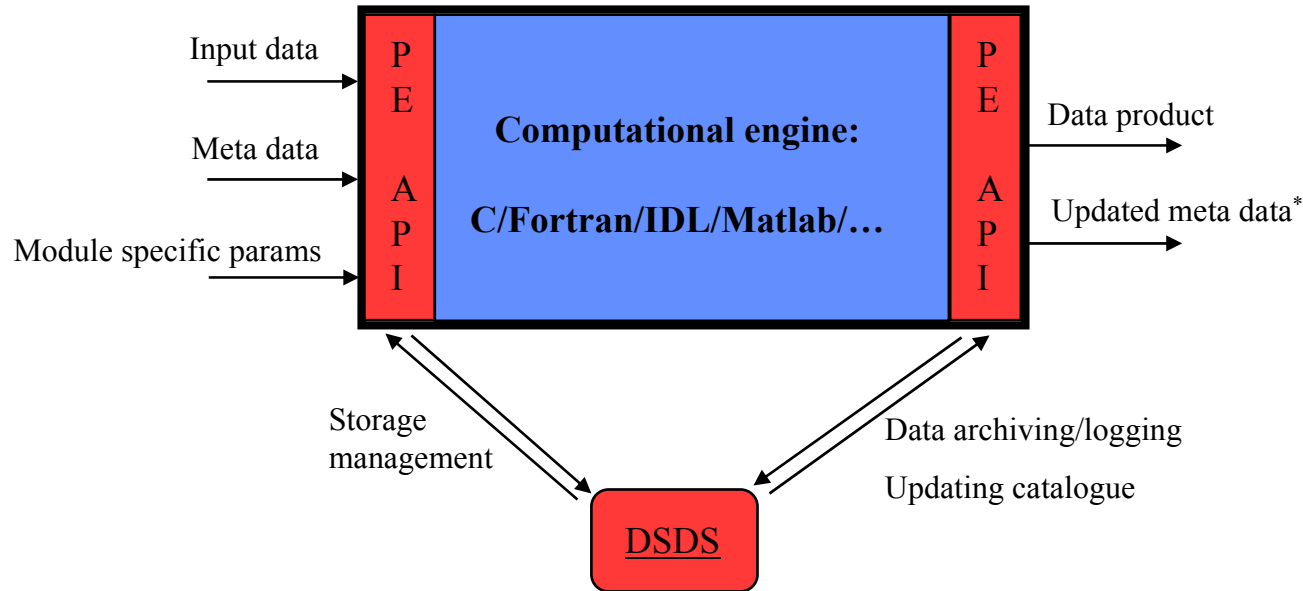


Module status and MDI heritage





Module structure



- **Design tasks**
 - **Identify intermediate and high-level data products desired by research community**
 - **Establish top level data flow and interface specs to**
 - › **Isolate module development from pipeline infrastructure**
 - › **Allow flexibility for evolving techniques (research codes)**
 - **Develop/import computational engines in HMI environment, verify:**
 - › **Correctness (test suites)**
 - › **Performance requirements (algorithm improvement, code tuning)**
 - › **Traceability, reproducibility (*: version & configuration info in meta data)**



Community contributions and collaboration

- **Contributions from co-I teams:**
 - Software for intermediate and high level analysis modules
 - Algorithm description (detailed enough to understand the contributed code)
 - Test data and intended results for verification
 - Time
 - › Explain algorithms and implementation
 - › Help with verification
 - › Collaborate on improvements if required (e.g. performance or maintainability)
- **Contributions from HMI team:**
 - Pipeline execution environment
 - Software resources (Development environment, libraries, tools)
 - Time
 - › Collaborate on defining module interface
 - › Help with porting code to target hardware
 - › Collaborate on algorithmic improvements, code tuning, parallelization
 - › Verification