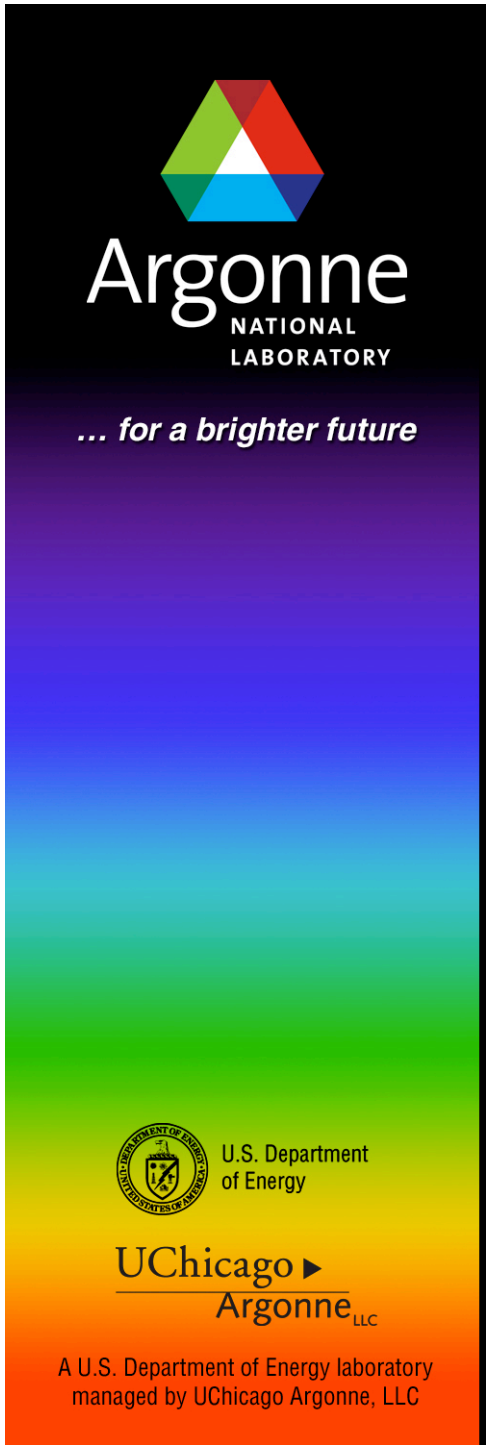


Role of Deputy Director for Code Architecture and Strategy for Integration of Advanced Computing R&D

Andrew Siegel

FSP Deputy Director for Code Architecture

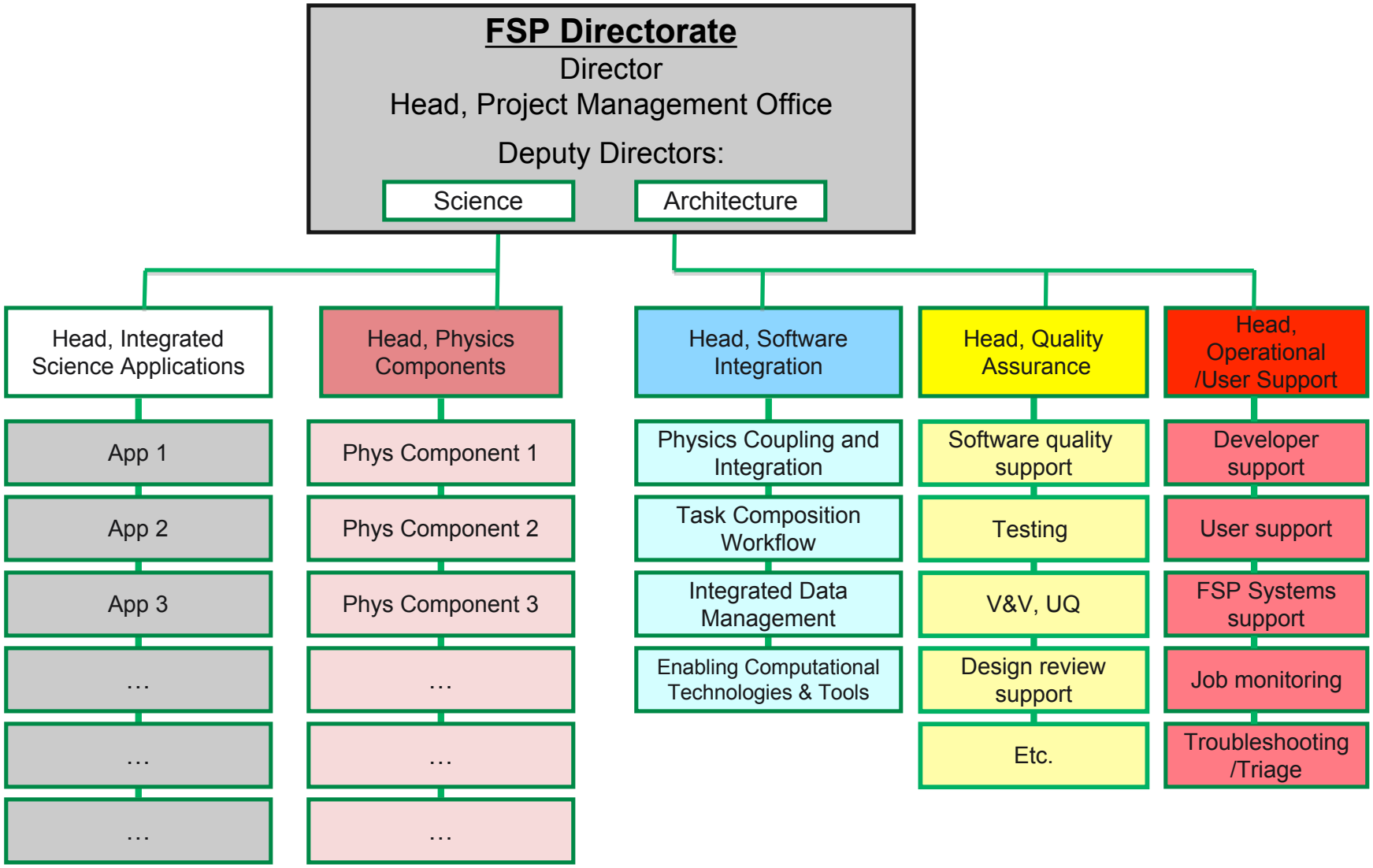
Outline

- Responsibilities of Deputy Director for Code Architecture
- High-level organization of FSP code development activities
- Strategy for incorporation of ASCR R&D
- FSP Computing architectures

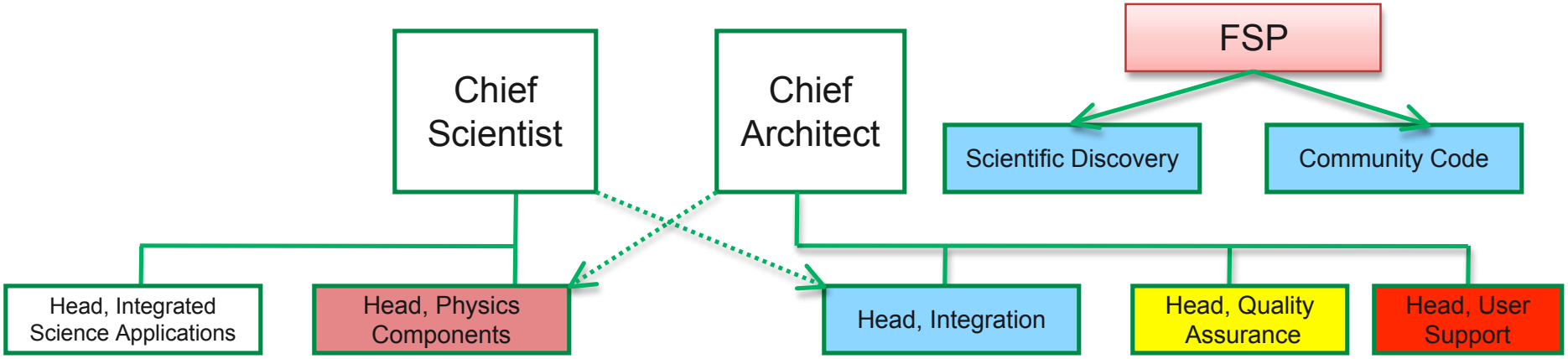
Outline

- Responsibilities of Deputy Director for Code Architecture
- High-level organization of FSP code development activities
- Strategy for incorporation of ASCR R&D
- FSP Computing architectures

FSP Management Organization Chart



Overall Responsibilities of Chief Architect



1. Oversees all aspects of code development
   - Software componentization of physics working with Chief Scientist
   - Software integration
   - Enabling computational technologies
   - Facilities support: user support, developer support, testing, etc.

1. Serves as ASCR Program Office contact for embedded applied math/CS Research & Development activities

1. Oversees strategy for migration of codes to future computing architectures

Responsibilities: Oversight of software integration

Deputy Director for Code Architecture:

- Constantly refines strategies to enable horizontal integration of FSP tools
  - application projects built off of common FSP physics components that form basis of integrated whole device model
  - application projects move toward common infrastructure/standards
  - application projects live under common software lifecyle
- Works with area leads to adopt processes that advance these goals without overly constraining integrated application teams in short-term.
- Leverage partnerships with ongoing DOE programs with track record of successful developments (e.g. Visit, Cubit, etc.)
- Combines knowledge of software architecture, physics, and numerics

Responsibilities: Oversight of User Support

- FSP will produce

- Process managed as relatively mature software process (integration talk)
  - Complex release, versioning, and repo management issues
  - Heavy emphasis on documentation
  - Provenance/pedigree …
  - Usability

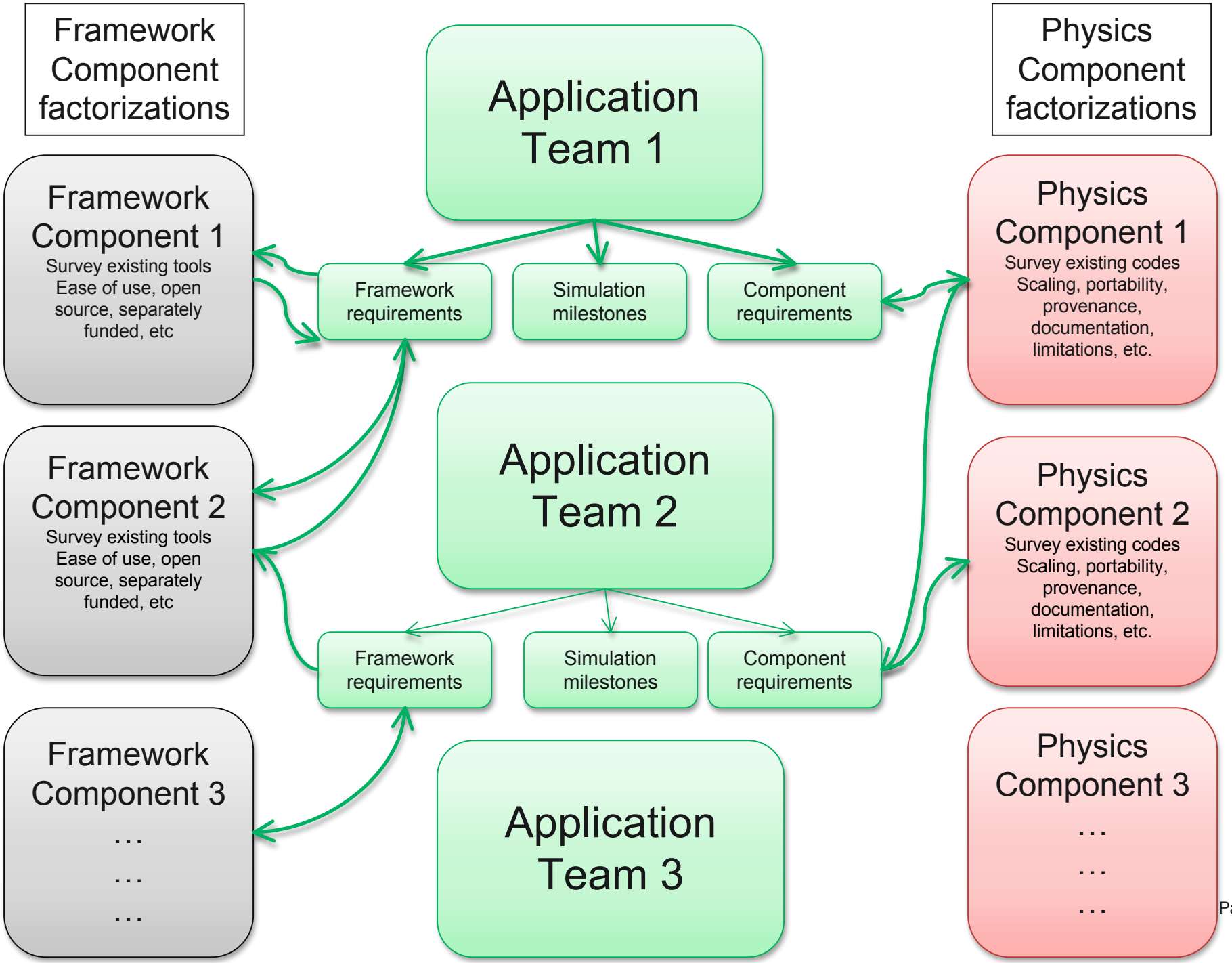
- Deputy Director oversees delicate balance to ensure that these processes work smoothly with overall integration and research goals

Outline

- Responsibilities of Deputy Director for Code Architecture
- High-level organization of FSP code development activities
- Strategy for incorporation of ASCR R&D
- FSP Computing architectures

Outline

- Responsibilities of Deputy Director for Code Architecture
- High-level organization of FSP code development activities
- Strategy for incorporation of ASCR R&D
- FSP Computing architectures


Framework Component factorizations
Application Team 1
Physics Component factorizations
Framework Component 1
Survey existing tools
Ease of use, open source, separately funded, etc
Framework requirements
Simulation milestones
Component requirements
Physics Component 1
Survey existing codes
Scaling, portability, provenance, documentation, limitations, etc.
Framework Component 2
Survey existing tools
Ease of use, open source, separately funded, etc
Application Team 2
Framework requirements
Simulation milestones
Component requirements
Physics Component 2
Survey existing codes
Scaling, portability, provenance, documentation, limitations, etc.
Framework Component 3
…
…
…
Application Team 3
Physics Component 3
…
…
…

Outline

- Responsibilities of Deputy Director for Code Architecture
- High-level organization of FSP code development activities
- Strategy for incorporation of ASCR R&D
- FSP Computing architectures

Partnership with ASCR

- ASCR supports research that dedicates efforts across disciplines to critical issues that underline simulation of complex systems.
- Significant progress in last decade working in "embedded mode" within application groups -- tools not built completely in the abstract
- Proto-FSPs modeled this way -- have examples of success in numerical methods, software engineering, visualization
- Must be careful not to separate embedded R&D topics with facilities support and technical expertise.

Areas of ASCR Research

- Innovative software engineering
  - CCA, ITAPS, …
- Numerical methods, algorithms, solvers
  - Apdec, TOPS
- Data analysis and Visualization
  - VACETS
- UQ and Stochastic Systems
- Mesh generation, mesh representation
  - ITAPS
- programming models/future architectures
  - Exascale Centers

How to move from general strategy to concrete to meet FSP research needs?

Process for incorporation of ASCR research

- In next stage of planning phase
  - Establish small crosscut team and require each application area to specify abstractly initial requirements for coupling, i/o, computing resources, physics components, etc. (end of September)
  - Key finding from proto-FSP review: “interdisciplinary teams required in each major management area …”
  - Translate into concrete embedded ASCR R&D activities
    - *enabling technologies: meshing, visualization, software coupling, etc.*
    - *linear/non-linear solvers, coupling methods*
- At implementation phase
  - Require that appropriate individuals/teams are brought on board to execute apps project.
  - Give integrated application leads flexibility to choose established collaborators
  - Allow application teams to bootstrap existing tools (e.g. from proto-FSPs).

Some issues raised Proto-FSPs: Applied Math

- Extending governing equations into new regimes of applicability
  - e.g. gyrokinetic equations into the edge
- Hybrid algorithms
  - e.g. kinetic models and continuum models through the pedestal
- Accuracy and stability of various coupling strategies
  - one-way coupling, boundary/interface coupling, operator-splittings
- Error estimation and uncertainty quantification for coupled models involving deterministic and Monte Carlo components and deterministic and statistical analysis; if we're coupling PIC in the core to a continuum model in the edge, how do we estimate the error in each and how do we combine these estimates to quantify the overall uncertainty?

Some issues raised with Proto-FSPs: CS

- portable workflow models
- data provenance capture
- different physics codes using different parallelism models interacting
- Blue Gene specific portability issues and impact on code architecture
- dynamic load balancing across and within components

Outline

- Responsibilities of Deputy Director for Code Architecture
- High-level organization of FSP code development activities
- Strategy for incorporation of ASCR R&D
- FSP Computing architectures

Existing and future HPC architectures

- Required computing resources dictated by science drivers
- Targeting a class of problems where physics fidelity benefits from next generation LCCs
  - Report: “Fusion Energy Sciences and the Role of Computing at the Extreme Scale”
- Leadership architectures rapidly moving away from one MPI process per core
  - MPI + ? Needed to take advantage of very high level node parallelism
  - Move from bulk synchronous to multitasking, etc. etc.
- Need to define software approach which ensures transition
  - Strong partnerships with exascale Center(s)