GENERATIONAL CORE
Background

Generational Core v. Base Runtime

- **Base Runtime**
  Provides hardware abstraction and defines basic, fairly stable system API and tools
  *E.g. glibc, libgcc, kernel*

- **Generational Core**
  Defines the generation of the operating system
  Includes Base Runtime as well as other, common shared libraries and stacks.
  Temporarily abandoned concept.
Implementation
It’s a module like any other

- Provides a stable, minimal, and bootable system
- Hardware abstraction layer
- Common libraries and utilities
- Minimal buildroot
- Container base image
- Inspired by LSB Core, POSIX userland, and Atomic Host
Development

And the challenges

- TIMTOWTDI; choosing content, minimization and splits
  - Keeping the API stable, minimizing the footprint and attack surface
- Building it, and building the building blocks
  - Dealing with packaging and FTBFS issues
- Unresponsive or opinionated package maintainers
- Development happens in parallel with the traditional release
  - Incorporating Rawhide changes into our frozen package set
Deliverables
Fedora 26

- PoC Base Runtime
  ... and its build environment
- System & container management modules
- A small selection of dynamic language runtimes and Fedora Server roles
- Project Boltron
- Unsupported, with no updates
- Made solely by the Modularity WG members
Deliverables
Fedora 27

- Base Runtime mostly stable
- Dynamic language runtimes mostly stable
- Automated builds & rebuilds
- Updates
- Modularity infrastructure open to public
- A new release of Fedora Server
- More content, hopefully
Deliverables
And beyond

- Fedora Cloud, Fedora Workstation
- Modular releases become our primary deliverable
- Traditional release still available
  But it is composed from modules
- Everything is awesome