

# SSD Firmware Development

Andy Tomlin

Aug 2012



# Firmware critical to SSD

- **Truism: Everybody knows that FW is critical to developing a SSD product**
- **What is not so clear is why is it so difficult, and how long does it take**
- **This is an attempt to provide some executive guidance for SW and non-SW leaders**
- **What does FW do**
  - ❑ Handle Host interface
  - ❑ Operate Flash management system
  - ❑ Control HW
  - ❑ Handle exceptions
  - ❑ Work around HW bugs and anomalous host behavior

# Some benchmark facts about SSD Firmware

## ■ How long does it take to develop a new code base

- ❑ From experience – minimum 2 years. If you think it takes less, it will take you longer.
- ❑ This can go up depending on your starting point
- ❑ Carefully consider the option of start from scratch vs leverage existing code base – do not automatically assume that a leveraged codebase is the right path – study carefully what the difference is between leveraged code vs new code base and the requirement for your product – particularly performance

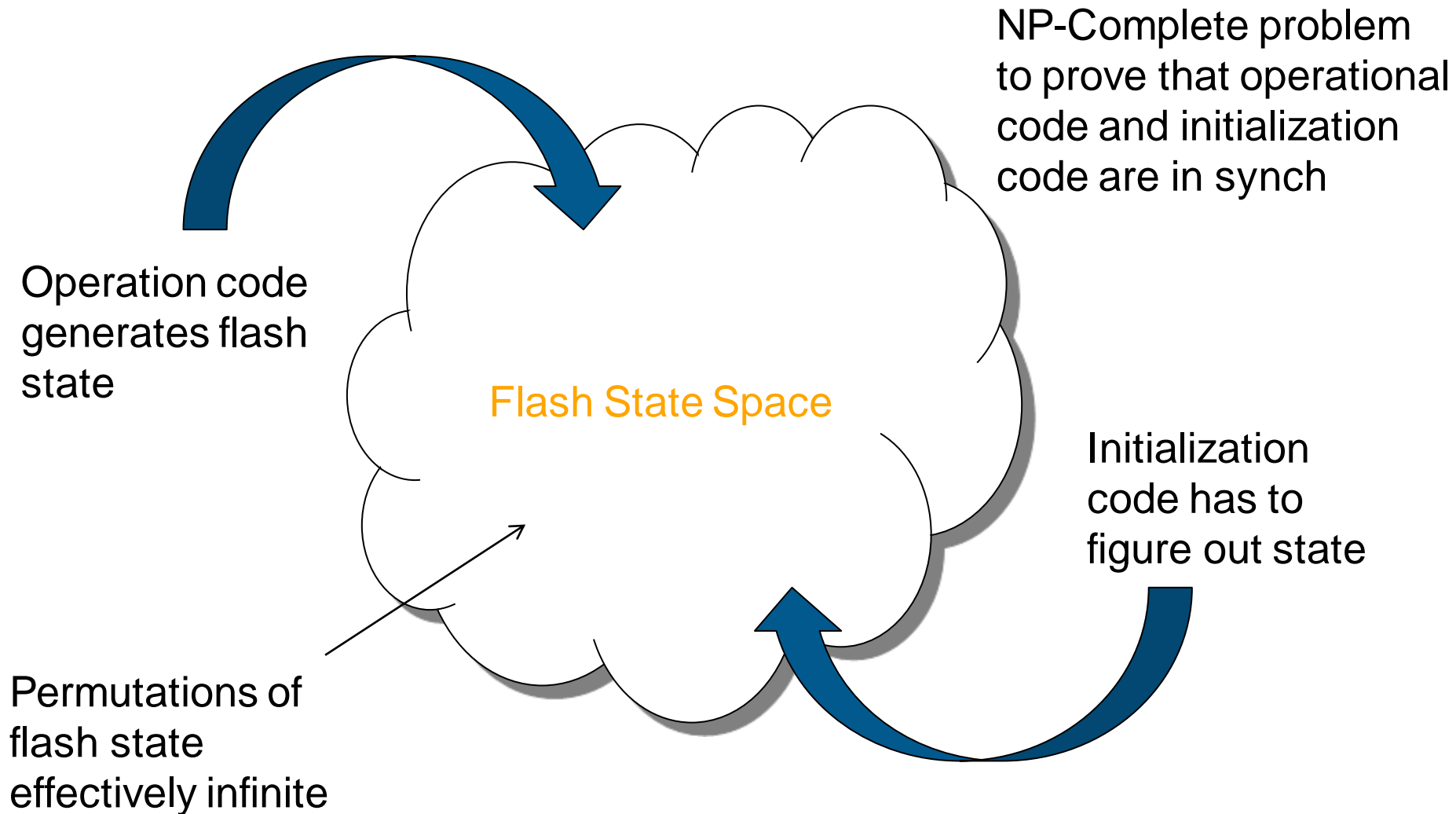
## ■ Performance

- ❑ CPU has to perform flash management functions and operate the hardware
- ❑ 10K CPU cycles per IOP is reasonable budget for this, for low levels of HW automation
- ❑ Multiple CPU's and or extensive HW automation is the only way to change the equation

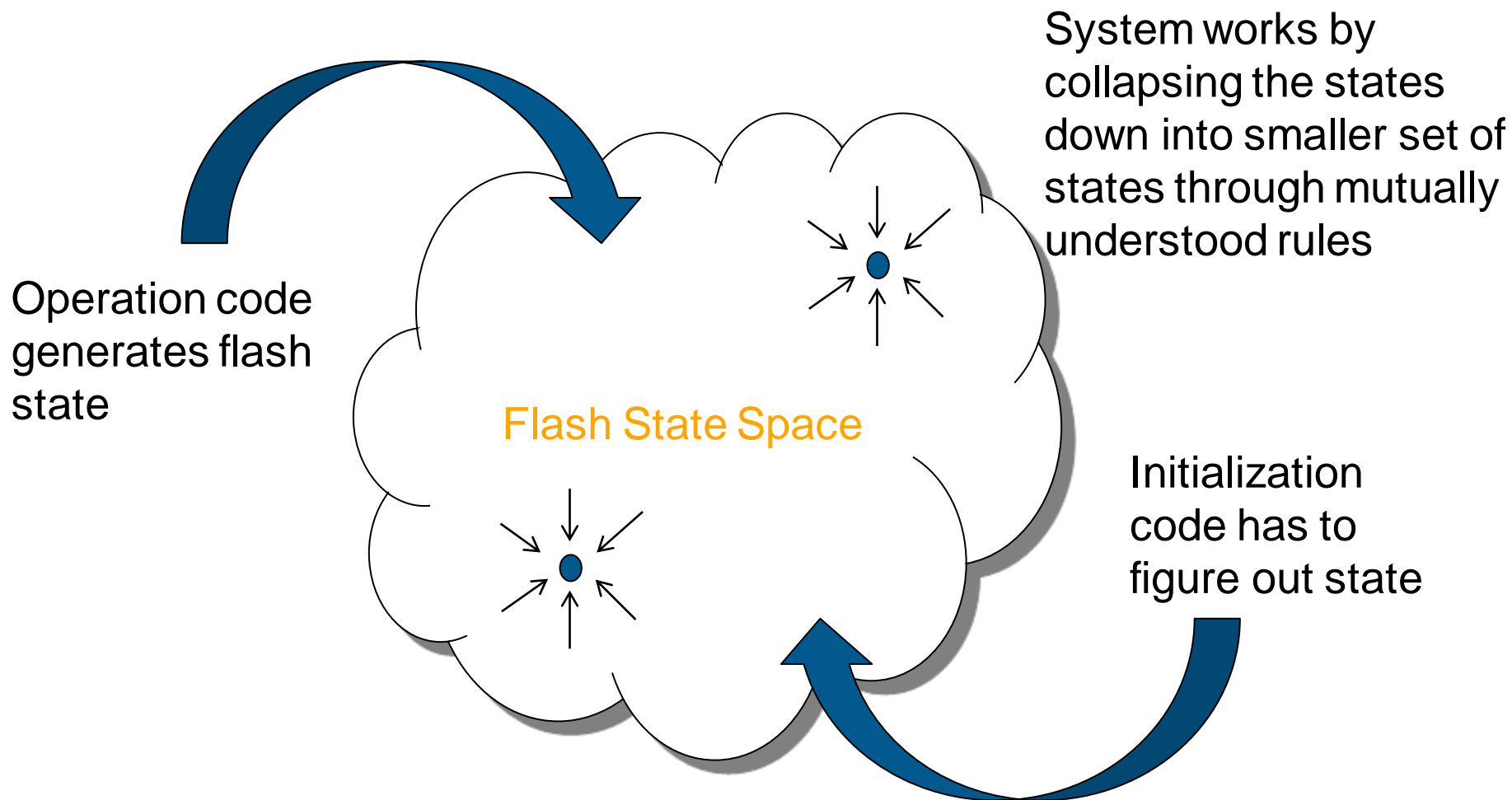
## ■ Power cycling

- ❑ A good benchmark for how long it takes to mature power cycling is 6 month from the point that it works on the Engineers bench
- ❑ Test infrastructure and modeling environment are keys to success

# Why is power cycling difficult?



# Why is power cycling difficult?



Can have 100% code coverage in both sets of code – but does not mean it works as the two pieces of code interface through infinite state space

# Keys to success

## ■ Test infrastructure

- ❑ Buy whatever you can, build whatever you cannot
- ❑ The quality of your product is defined by how well you test
- ❑ Make sure you can test at speed – SSD's are pushing the boundaries of storage performance and if you cannot stress them you are not testing them. Previous generation test platforms may need to be completely replaced
- ❑ Make sure your testers are 'Evil'

## ■ Do it right from the start

- ❑ Do not underestimate number of people or time it takes
- ❑ Power cycling will take longer to mature than you think
- ❑ Hire the best talent you can, collocate with HW developers if possible
- ❑ Utilize Software development best practices