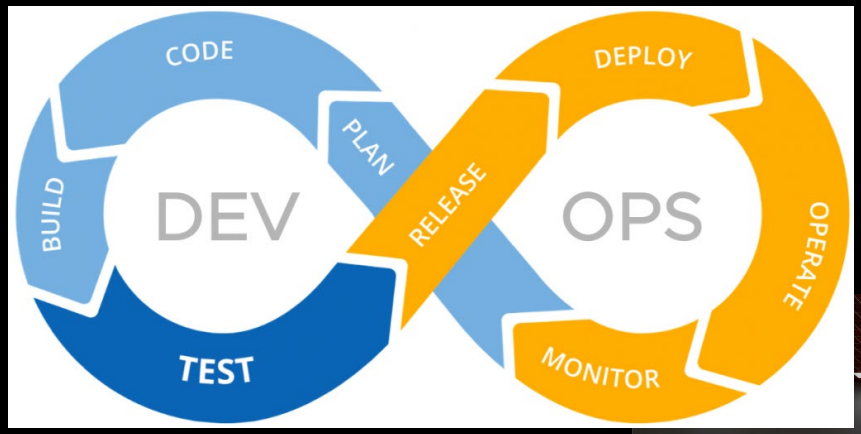
(U) Does DevOps Matter to the Warfighter?



The views expressed are those of the author and do not reflect the official policy or position of the US Air Force, Department of Defense, or the US Government.

Lt Col Christopher Hocking (SAF/AQ)
Materiel Leader, Weapon System Integration

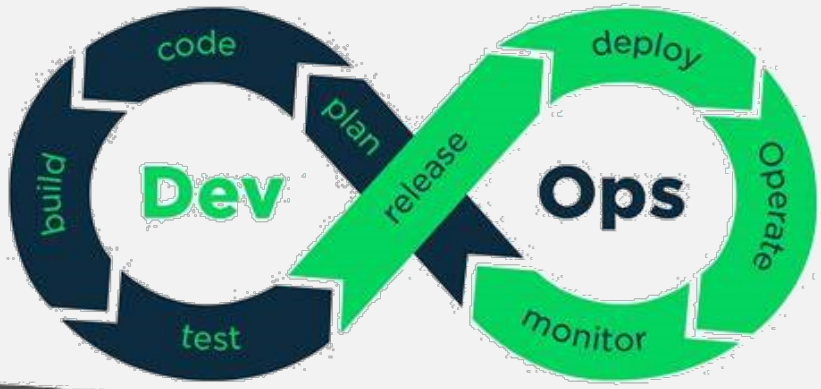
How do we deliver capability?

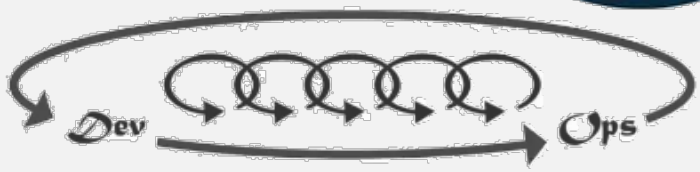
- **Systems Thinking**. The output of the acquisitions process is a product or capability, every investment, action, or technology should be assessed in view of delivering that capability
- **Experimentation and Learning**. Program Office is a learning culture, document, share, shift away from policy compliance and toward iteration
- **Feedback**. Reward risk taking, encourage focus on delivery over policy or process
- **Flow**. Work only moves forward, break complex tasks into manageable pieces of effort

DevOps Philosophy

Three Ways:

1. **Flow/Systems Thinking**. Smooth/continuous flow of work from development to production.
2. **Feedback/Amplify Feedback Loops**. Fast and effective feedback loops.
3. **Continual Learning/Experimentation**. Culture of continuous improvement and learning.





What do we value?

- Speed. Tight cycle times enables user feedback; reduces integration risk
- Quality. Capability matches **user's** needs while minimizing defects
- Focus. Deliver fewer working capabilities vs. many partial capabilities
- Collaboration. Synchronize efforts of Government, prime, and suppliers

Objective: Field effective warfighting capabilities at the speed of relevance

“Not all software is the same…”

DevOps: produce a better product, faster by building Minimum Viable Products (MVP) early and often.
Maximize similarity between development and operational environment.

“Dev**Ops**”…Ops = {pilot, maintainer, mission planner, adversary}?

Microsoft Office →
AU Blackboard →
Kessel Run →
F-22 Aircraft →

DIB SWAP (3 May 2019)
“…not all software is the same”
- *Type A: COTS*
- *Type B: Customized Software*
- *Type C: COTS HW/Operating Systems*
- *Type D: Custom Software/Hardware*

← DevSecOps “apps”
← Dev*Ops (OFP)

} Need a change agent to improve program outcomes and enterprise

Industrial DevOps: DevOps philosophy for complex, embedded systems. In alignment with modern software processes used by flight/safety-critical industry. Key challenges:

- Cyber security
- Embedded hardware (unable to fully test in “cloud”)
- Airworthiness
- Safety
- Nuclear Surety
- Safe Weapons Separation

Without a “silver bullet” software approach, what do we use to measure success?

Dev*Ops

I highly recommend everyone read this report! →

DIB SWAP (3 May 2019)
"…not all software is the same"
- *Type A: COTS*
- *Type B: Customized Software*
- *Type C: COTS HW/Operating Systems*
- *Type D: Custom Software/Hardware*

Continuous Development *Hours - Days*	T_K = (Dev)	
Continuous Integration *2 weeks – 1 month*	T_{K+1} = (Dev)(Ops)	← **Industry** culture and policy evolved; blurred lines between dev and ops environments
Continuous ATO *+ hours*	T_{K+2} = (Dev)(Sec)(Ops)	← **Security** culture and policy evolved; baked into automated pipeline
Continuous Safety *+ months*	T_{K+3} = (Dev)(Sec)(Safety)(Ops)	
Continuous Airworthiness *+ months*	T_{K+4} = (Dev)(Sec)(Safety)(AW)(Ops)	
Continuous Test *+ years*	T_{K+5} = (Dev)(Sec)(Safety)(AW)(DT/OT)(Ops)	
Continuous Weapons *+ months*	T_{K+6} = (Dev)(Sec)(Safety)(AW)(DT/OT)(Seek Eagle)(Ops)	
Continuous Nuclear *+ months*	T_{K+7} = (Dev)(Sec)(Safety)(AW)(DT/OT)(Seek Eagle)(Nuc)(Ops)	
…	T_{K+n} = (Dev)*(Ops)	

} Must consider and evolve these elements as well to enable more automated pipeline!

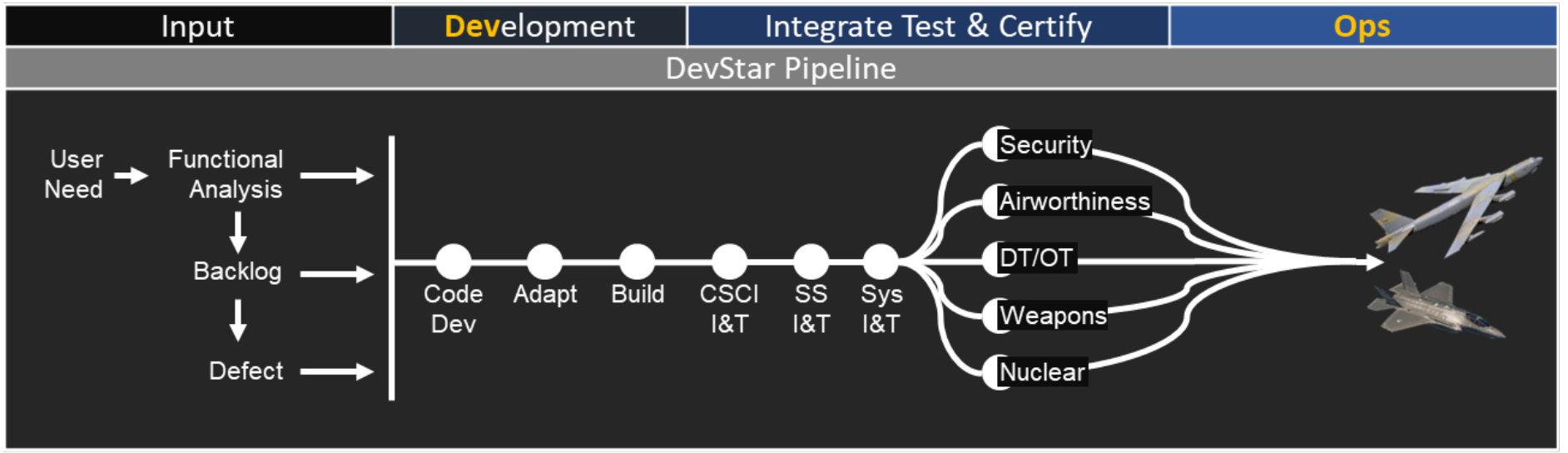
Keys to Success
1. Evolve culture and policy
2. Early stakeholder involvement
3. Automated vs. manual process

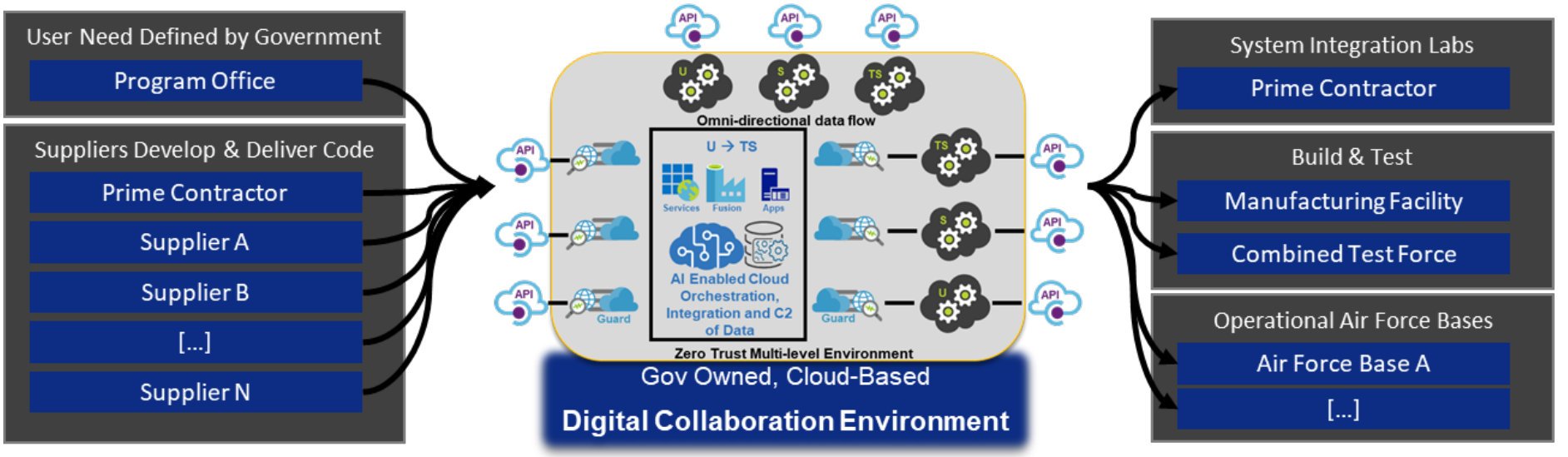
Embedded weapon systems require DevStar (Dev*Ops) but rate limits still exist!

Dev*Ops Philosophy

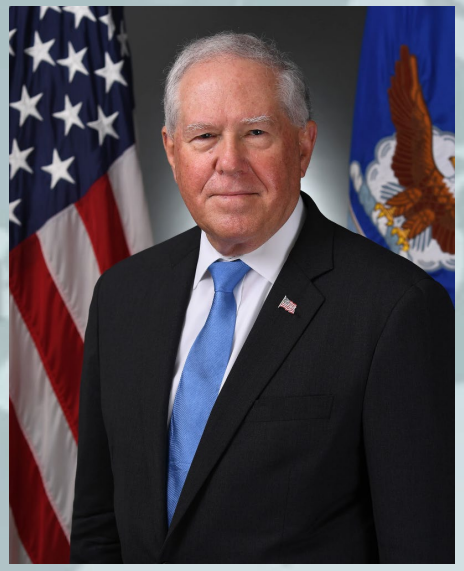
1. Stable requirements and funding
2. Robust integration and test capacity
3. Deliver working code. Fast.
4. Shift DT/OT left
5. Embrace reuse
6. Adopt open architecture
7. Employ software factory: Dev*Ops
8. Embed organic sustainment early
9. Model Based Systems Engineering
10. Metrics that matter

Missionized Dev*Ops





"[My priorities] in order … are **China, China, and China**."



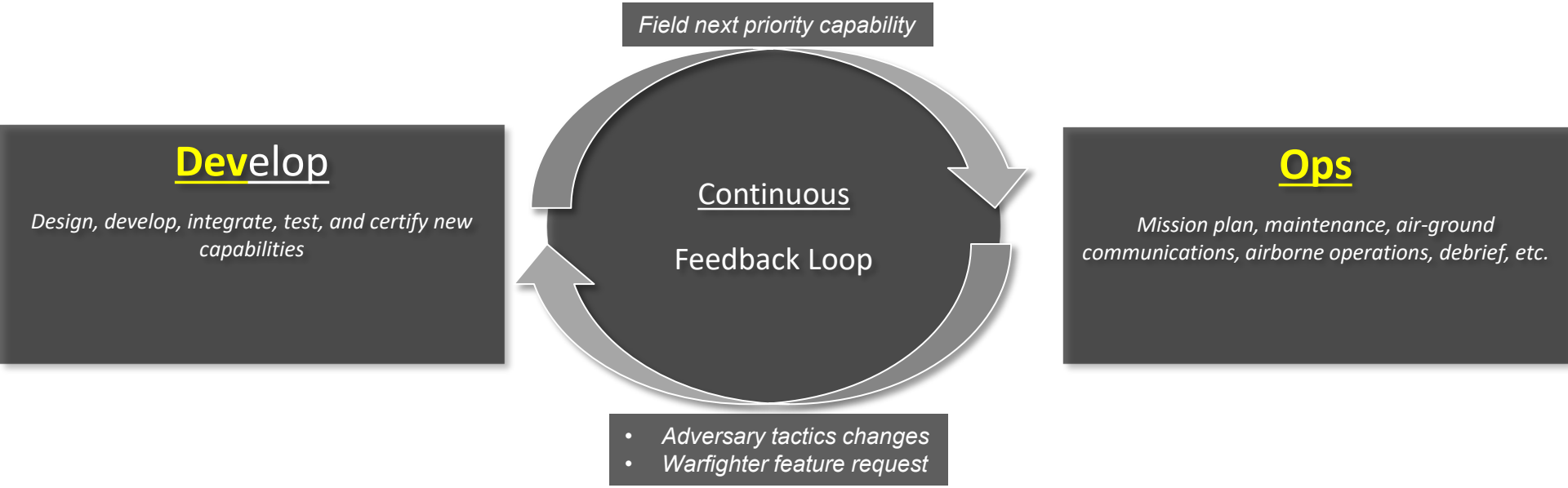
HON Frank Kendall

Gen C.Q. Brown, CJCS

"We have a window of opportunity. **[We] must accelerate change** to control and exploit the air domain to the standard the Nation expects and requires from us. **If we don't change** – if we fail to adapt – we risk losing the certainty with which we have defended our national interests for decades. **We risk losing a high-end fight.**"

What complacency and overmatch looked like in 1991…

What (IT) does it take to field a modern weapon system?

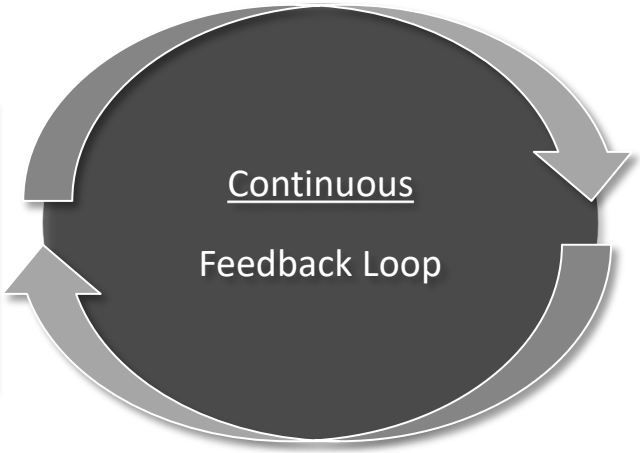


Success belongs to the side who iterates and learns more quickly.
Those who do not will find themselves trapped in a negative spiral of paralysis.

What (IT) does it take to field a modern weapon system?

Develop

Design, develop, integrate, test, and certify new capabilities



Ops

Mission plan, maintenance, air-ground communications, airborne operations, debrief, etc.

Continuum of Weapon System IT

- Collaborative environment with USG, prime, & suppliers
- DevSecOps pipelines
- Digital engineering tools

Data: Near real time ingest, analyze, transport, and store large amounts of data (petabyte scale)

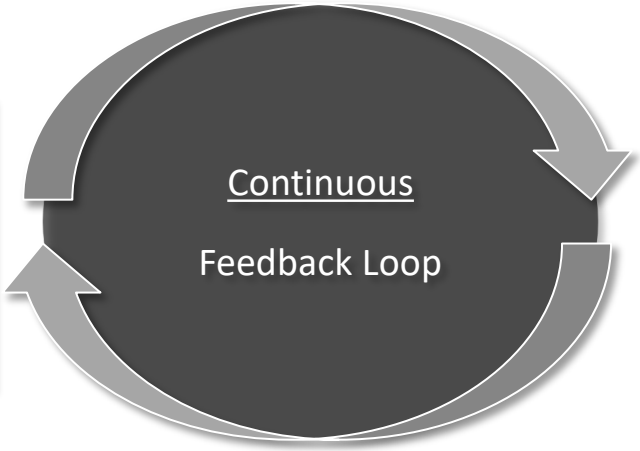
- Common interfaces to compose real-time system-of-systems kill chains
- Expose data to intel & MX
- Collaborate mission plan & execution
- Disconnected operations

Success belongs to the side who iterates and learns more quickly.
Those who do not will find themselves trapped in a negative spiral of paralysis.

What (IT) does it take to field a modern weapon system?

Develop

Design, develop, integrate, test, and certify new capabilities



Ops

Mission plan, maintenance, air-ground communications, airborne operations, debrief, etc.

Continuum of Weapon System IT

BizAdmin	Dev & Integrate	Test	Production	Ops
Administer program, contract financials, and interchange with industry	*Technical work to design weapon system, including hardware and software*	*Ground and flight test (DT/OT)*	*Physically manufacture the weapon system*	*Capabilities to operate, employ, and sustain the weapon system*

Success belongs to the side who iterates and learns more quickly.
Those who do not will find themselves trapped in a negative spiral of paralysis.

Day in the life of weapon system ops…

Pre Mission


Weapons Planning


Mission Planning


Maintenance Actions

In Mission


Common Operating Picture

Post Mission


Intel & Mission Data Exploitation


Mission Debrief

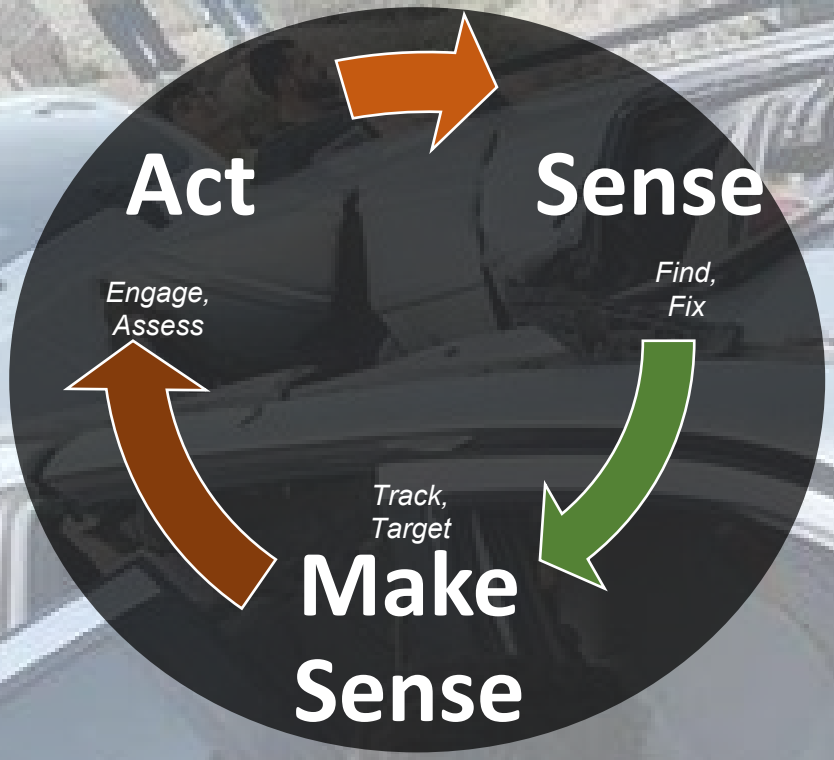

Maintenance Debrief

Digital infrastructure is required to launch, update, and operate all modern weapon systems.
It isn't just a support element; it is an integral part of the weapon system kill chain.

Success belongs to the side who iterates and learns more quickly.
Those who do not will find themselves trapped in a negative spiral of paralysis.


One successful kill chain…
© Twitter / Obretix

Kill Chain 101: Creating Operational Effects



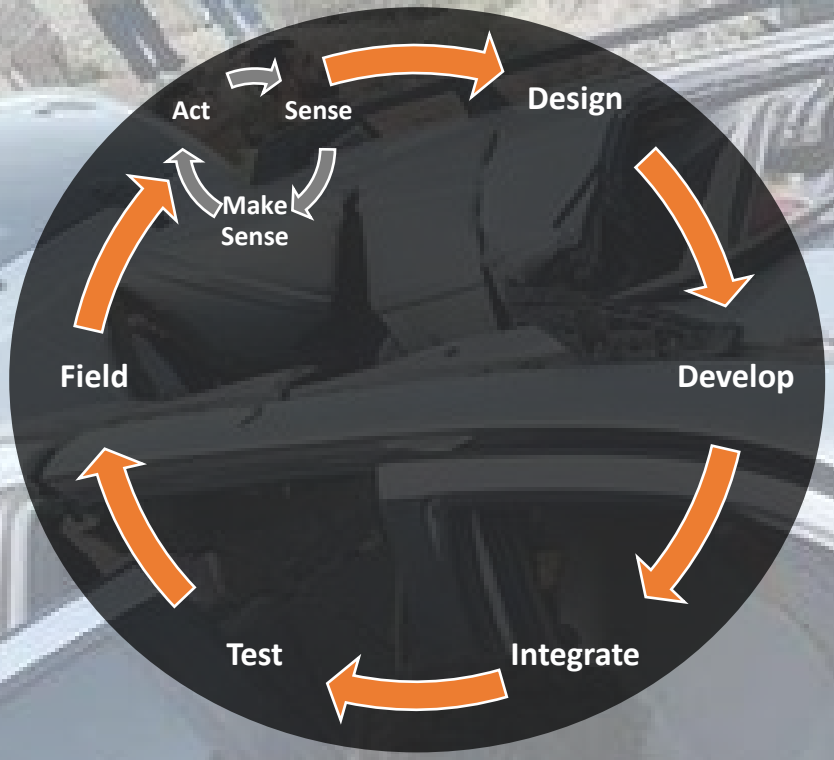
Kill-Chain Observations:

1. Requires full connectivity through F2T2EA stages
2. Underpinned by robust digital infrastructure
3. Currently heavily man-in-the-loop processes
4. Adversary "breaks" it denying C2/computation

Adversary gets a vote!

How do we close kill chain when red tactics changes require a new blue capability?

Kill Chain 101: Creating Operational Effects



Kill Chain Observations:

1. Requires full connectivity through F2T2EA stages
2. Underpinned by robust digital infrastructure
3. Currently heavily man-in-the-loop processes
4. Adversary "breaks" it denying C2/computation

***Expanded* Kill Chain Observations:**

5. Digital infrastructure expands beyond ops to software development, digital engineering, and BizAdmin use cases
6. Adversary will attempt to disrupt this kill chain by cyber espionage or attack

End-to-end kill chain includes full feedback loop between "Dev" and "Ops"!

Recommendations: Flow Accelerators

1) Open Architecture

- **Open Architectures**. Compose systems using open standards and interfaces, avoid re-engineering interfaces, build once, deploy many times. Simplicity = speed.
- **Mandate end-states not technical approaches**. Focus on "MVP" government reference implementations that balance industry agility against data rights
- **Digital Infrastructure as 1st Class Citizen**. Fund and organize around digital infrastructure as an essential weapon system capability.
- **Connect**. Make it quick/easy/cheap to connect the full digital thread. Include full lifecycle (development through operations) with all stakeholders, including suppliers, prime, and government.

…expose data by default for ops, intelligence, and C2 community to leverage.

2) Digital Engineering

3) Agile Software Dev

Align people & process to use modern commercial tech-driven solutions…to deliver effective mission systems

Recommendations: Decision Accelerators

- **Democratize Data**. Publish data sources by exposing interfaces, connect high value data kill chains, enable operators, at every level, to fuse their own information
- **Realign.** No database calls, not direct connections, all data shared via Application Programming Interfaces (API), refactor applications to take advantage of SAP cloud capabilities
- **Catalog**. Document critical weapon system data, publish a data catalog on the most appropriate enterprise networks
- **Expose**. Share critical data from relevant systems using APIs and federated identity
- **Converge**. Drive critical decision data to the right user in the right timelines using commercial content delivery models

Without seamless information exchanges fusing the contested environment fight, we increase risk to the warfighter

Recommendations: Building a DevOps Culture

- Read. Kill Chain, DIB SWAP, Phoenix Project, Unicorn Project, DevOps Handbook
- Focus. Kill buzzword bingo/innovation theater. Look at every action; pivot if not moving operational capability needle.
- Agility. Bake in agility to HW/SW architecture to be responsive evolving needs.
- Metrics. All metrics are proxies. Fielded capability is only measure that matters.
- Interoperable. Effective mission systems won't work in information silos.
- Action. Bias for action & ownership
- Are you comfortable / constantly succeeding? You're not pushing hard enough.
- Self Reflection. Continuous learning…not a self-licking ice cream cone
- Relentless focus to deliver customer value. **But who is our end user?**

Gen C.Q. Brown

"You're **not** going to **accelerate change** and **be comfortable** at the same time."

Mission Agility: What does “right” look like?

Respond to a dynamic threat environment change within hours to days…not years to decades

What complacency and overmatch looked like in 2022…