beeckcenter
social impact + innovation

FEBRUARY 2023

A CASE STUDY IN CONSORTIUM-BASED, OPEN-SOURCE, INTERGOVERNMENTAL SOFTWARE COLLABORATION

Architecting Effective Governance in ISCs

TABLE OF CONTENTS

About the Beeck Center for Social Impact + Innovation

The Beeck Center is an experiential hub at Georgetown University that trains students and incubates scalable, leading edge ideas for social change. We believe impact at scale requires the courage to think and behave differently. Our work centers on investing in outcomes for individuals and society. We equip future global leaders with the mindset to promote outcome-driven solutions, using the tools of design, data, technology, and innovation. We convene actors across the public, private, and civic sectors to advance new tools, frameworks, and approaches necessary to achieve these outcomes.

About the author

Colleen Pulawski is a researcher for the Digital Service Network and the Intergovernmental Software Collaborative at the Beeck Center for Social Impact + Innovation at Georgetown University.

Acknowledgements

The author would like to acknowledge the ActivitySim consortium, and in particular the members who generously dedicated their time in interviews, for its engagement with this research process.

This research would not have been possible without the support of other team members on the Beeck Center's Intergovernmental Software Collaborative project. Many thanks to student analyst Abbey Swartzwelder who supported document compilation and review, Community Manager Nalani Saito who supported editing and interview transcription, and former Fellow Shelby Switzer who counseled on key technical matters and offered essential thought partnership throughout the research process.

Executive Summary

The Beeck Center believes that intergovernmental software collaboratives (ISC) have the potential to address longstanding and seemingly intractable issues in government IT projects. When done right, ISCs can help save governments time and money while increasing the likelihood of success of the project, the quality of the end solution, and satisfaction ratings from the public. However, the journey to success is rife with challenges which threaten the sustainability of ISCs.

These challenges have contextual dimensions which necessitate contextual solutions. This context specificity means that guidance on an issue-by-issue basis (e.g., "here's how to navigate [fill-in-the-blank-issue]") may be only one part of the puzzle in supporting ISCs. Meanwhile, governance — defined as the means of achieving the direction, control, and coordination of multiple, independent government organizations on behalf of a software solution which they jointly develop and maintain — offers a broad frame for understanding and learning from how ISCs structure their ability to respond to the unique challenges they face.

One of our goals at the Beeck Center for Social Impact + Innovation is to facilitate the emergence and sustainability of more ISCs as we strive to make them the default for software development in government. We believe fostering a better understanding of ISC governance will help our team create more generalizable and scalable advice for practitioners compared to issue-specific guidance. To understand governance more deeply, we developed a set of evergreen research questions focused on the design, implementation, and responsiveness of governance in ISCs. We then conducted an initial case study to help us start answering those questions.

This case study examines the governance of ActivitySim — an open-source, activity-based travel behavior modeling tool developed and managed by a consortium-based ISC. ActivitySim launched over eight years ago and has an extensive, publicly-defined governance infrastructure which shapes collaboration between members of the consortium. For the last several years, we have looked to ActivitySim as a strong example to mine for lessons in governance.

Six lessons in governance for ISCs

Studying ActivitySim offered six actionable lessons on achieving clear, effective governance for ISCs:

1. A dynamic approach to governance helps ISCs adaptively meet challenges as they mature.
2. A balance of formal and informal governance strategies promotes clarity and helps ISCs respond to diverse challenges.
3. Intentionally governing member growth is more important than starting small.
4. Distributing leadership responsibilities and planning for succession promotes sustainable and resilient leadership.
5. Governance can help ISC members remain empowered in vendor relationships.
6. Frequent touchpoints mitigate the potential for severe conflict.

These lessons iterate on the guidance offered in our previous research, including how the degree of formality in governance matters, how ISCs should start and grow their numbers, and how highly centralized leadership can function as a point of vulnerability.

What remains clear: governance is indeed a crucial determinant of an ISC's success. Government practitioners who work in ISCs and dedicate the time, energy, and resources to governance that it deserves will be well-positioned to foster smooth collaborations that produce effective shared software solutions.

Read Our Summary

For a short summary of the six lessons in governance for ISCs, read our blog post about this research.

Defining governance in ISCs

Governance in software development has rarely been defined with specificity, and scholars haven't settled on a standard definition. In ISCs, we can define governance as the means of achieving the direction, control, and coordination of multiple, independent government organizations on behalf of a software solution which they jointly develop and maintain.[1]

This broad definition allows us to explore a number of structures, rules, processes, and norms that, together, shape outcomes in ISCs; for example software licenses, role definitions, documentation practices, code contribution and review processes, meeting types, and decision-making authorities.

Motivation for studying governance in ISCs

Intergovernmental software collaboratives (ISC) — two or more government agencies jointly supporting the development of software for their collective use, operating under some kind of a governance structure — have the potential to address longstanding and seemingly intractable issues in government IT projects. When done right, ISCs can help save governments time and money while increasing the likelihood of success of the project, the quality of the end solution, and satisfaction ratings from the public.

But the journey to success is rife with challenges which threaten the sustainability of ISCs, including: different motivations, goals, and attitudes among ISC members; no clear precedent for working across autonomous organizations, uncertainties about decision-making authority; and asymmetries in key areas of expertise, technical or otherwise.

These challenges have contextual dimensions which necessitate contextual solutions. This context-specificity means that guidance on an issue-by-issue basis (e.g., "here's how to navigate [fill-in-the-blank-issue]") may be only part of the puzzle in supporting ISCs. Meanwhile, governance — defined as the means of achieving the direction, control, and coordination of multiple, independent government organizations on behalf of a software solution which they jointly develop and maintain — offers a broad frame for understanding and learning from how ISCs structure their ability to respond to the unique challenges they face.

One of our goals at the Beeck Center for Social Impact + Innovation is to facilitate the emergence of ISCs as the default for software development in government. We believe fostering a better understanding of ISC governance will help our team create more generalizable and scalable advice for practitioners compared to issue-specific guidance.

Research questions

Our previous research suggests that the way an ISC navigates hazards and achieves sustainability is strongly influenced by the clarity of its governance. While we know clear governance is a prerequisite for successful ISCs, there's still much to learn about exactly what "clear" governance means, and what other traits might make for governance that is equipped to handle the challenges ISCs face. To better unpack governance in ISCs, we developed a set of evergreen research questions aimed at uncovering lessons that existing and aspiring ISCs could apply in their work. We organized these questions around the design, implementation, and responsiveness of governance:

Governance design: What strategies might ISCs deploy to arrive at an initial governance infrastructure? What tradeoffs must be navigated in various approaches? How might ISCs productively mediate conflicts or competing needs, attitudes, and goals when designing governance?

---

1. This definition is adapted from M. Lynn Markus' definition of governance in open source software (OSS), which was informed by an earlier definition of governance for public administration.

Governance implementation: How might ISC's governance models influence the software solutions they develop? How might ISC's governance models affect the way they navigate challenges which arise as they mature; for example, the efficiency, efficacy, adaptability, and localizability of solutions, shifting maintenance loads, the addition of new members, public interaction with a codebase, or the ability to sustain funding?

Governance responsiveness: How might ISC's governance models be designed to support members in anticipating and navigating future trade-offs — known or unknown; for example, a growing membership base, emergent privacy and security concerns, tensions between local versus aggregate utility, or shifting funding sources?

As we noted above, challenges faced by ISCs are highly contextual. This means there's much to learn from examining the concrete ways that governance is designed, implemented, and responsive in the specific context of particular ISCs. Accordingly, we looked at a real-world example to help us uncover answers to these questions. This case study examines the governance of ActivitySim — an open-source, activity-based travel behavior modeling tool developed and managed by a consortium-based ISC. ActivitySim launched over eight years ago and has an extensive, publicly-defined governance infrastructure which shapes collaboration between members of the consortium. For the last several years, the Beeck Center has looked to ActivitySim as a strong example to mine for lessons in governance.

Introduction to ActivitySim

ActivitySim is an advanced, open-source, activity-based travel behavior modeling tool that is collaboratively developed by a consortium of metropolitan planning organizations (MPOs) and state-level transportation planning agencies from across the United States. The project launched after staff from three MPOs had the idea to pool their funds to support the joint development of open-source ABM software and share learnings and maintenance.

In 2014, they founded the ActivitySim consortium, which quickly grew to five MPOs. Today, the consortium includes twelve agencies, two of which are at the state level, and remains open to new members.

Membership has grown steadily since ActivitySim's inception. Today, ActivitySim is currently in use or under development by at least a dozen government agencies and three universities in the United States. Over the course of its seven years of existence, the consortium has navigated challenges not only in the complex field of ABM software development and sharing, but also those common in cross-organizational collaboration such as the need to reconcile competing needs or overcome legal hurdles.

Overview of activity-based travel demand modeling software in government

Travel demand models (TDM) generate information about the impacts of alternative transportation and land use investments and policies, as well as demographic and economic trends. Activity-based models (ABM) such as ActivitySim are the most advanced form of TDM, replicating traveler decisions more accurately than other modeling approaches. This is because ABMs better reflect the constraints of time and space, the links between activities and travel for both individuals and households, and information about individual- and household-level attributes. TDMs in general provide key inputs for designing and implementing resilient, sustainable, and safe transportation systems. ABMs can help support more accurate, sensitive, and equitable scenario planning for transportation planners and policymakers.

But ABM implementation costs tend to be high, particularly for smaller governments. Historically, agencies have tried to reduce these costs by customizing ABMs shared by other governments in an ad-hoc way, though this still tends to mean high maintenance costs and difficulties sharing back improvements. ActivitySim's collaborative approach to development addresses these core challenges and has matured the practice of sharing ABM software in government.

Research approach

In our initial research on software sharing in government, the Beeck Center recommended ActivitySim's governance documentation as a template for other ISCs to emulate. In 2022, we examined ActivitySim's governance infrastructure in depth to better understand how it functioned in practice and identify generalizable lessons in ISC governance. Over the span of four months, we:

+ **Conducted nine interviews with ActivitySim stakeholders**, including individuals from member organizations of the ActivitySim consortium and the project's externally contracted developers.
+ **Observed four meetings of the consortium**, including member-vendor and member-only meetings, which focused both on current and future development.
+ **Compiled and reviewed 160 documents from ActivitySim's GitHub repository**, including both technical and process documentation spanning seven years of development.

Overview of ActivitySim's governance model

The most notable and actively used elements of ActivitySim's governance model include:

Roles

+ **Developers**: Developers contribute code or documentation to the ActivitySim project. In theory, anyone can be a developer, not just those in the **PMC** or the **Bench vendors** (see below). In practice, third-party developers do not play a major role in the ActivitySim project.
+ **Committers**: Committers are given write access to the ActivitySim codebase and are required to sign a **Contributor License Agreement** (see below) to gain this status.
+ **Project Management Committee members (PMC):** The PMC is responsible for guiding the overall direction of the project, including running the informal request for proposal (**i-RFP, see below**) process and allocating funds to **Bench vendors** (see below). The committee is currently comprised of people who work in governments that fund the ActivitySim project. Membership in the PMC is by consensus invitation, though external governments can request to join. The PMC is led by a **Chair** and a **Vice Chair**, which are elected, non-term-limited positions. PMC members can also be **Committers**.
+ **Bench vendors**: At present, ActivitySim is developed primarily by a bench of three vendors. Each year, these vendors are asked by the **PMC** to submit to the **i-RFP**. Vendors are not required to submit to the i-RFP, nor is the PMC required to contract each vendor in a given year. **Bench vendors** are also **Committers**.
+ **Project manager:** Each year, the PMC allocates funding to one **Bench vendor** to assign a project manager responsible for coordinating development work. The project manager is responsible for working with all Bench vendors to ensure work meets stated goals.
+ **Association of Metropolitan Planning Organizations (AMPO)**: AMPO is a collaborative network of MPOs that focuses on peer-exchange and knowledge building. It provides centralized project management and institutional capacity for ActivitySim that is often difficult to achieve in large, cross-organizational collaborations. Early PMC members considered AMPO a neutral party that was relatively well-suited to stewarding particular governance structures, such as the project's **MOA**s (see below) as ActivitySim's collaboration got off the ground.

Processes

+ **Annual Scoping**: Each year, the **PMC Chair** and **Vice Chair** lead a formal process to prioritize **PMC member** needs and goals and make funding allocation decisions. Members are invited to submit desired development work, rank submissions from others, estimate the level of effort required for each submission in dollars, and propose an allocation of their $35,000 pooled fund contribution across various work activities. The process results in a proposed **Scope of Work** for each annual phase of the ActivitySim project. The Scope of Work summarizes the PMC's collectively-determined priorities for the year.
+ **Informal requests for proposals (i-RFP):** Based on the **Scope of Work**, the **PMC** asks the project's bench vendors to submit proposals detailing how they'd approach the scoped priorities. The process is informal and only open to **Bench vendors**. Based on the proposals, the PMC makes final decisions about the amount of funds to allocate to each vendor and which work they will be responsible for.
+ **Weekly meetings:** The PMC meets with vendors twice a week throughout the calendar year. These meetings typically focus on short- and mid-term development. They function as a vital communication channel between the PMC and vendors since granular questions about development are addressed, agile development is promoted, user needs are surfaced, and plans for review and testing are solidified.
+ **Monthly partners-only meeting:** In addition to the twice-weekly meeting, the **PMC** holds a members-only meeting once per month where longer-term planning or sensitive topics can be discussed without vendor influence. This is the primary space where the **Annual Scoping** is discussed.

Assets

+ **Memorandum of Agreement** (**MOA**): To join the PMC, members are required to sign an MOA, which commits them to an annual pooled fund contribution of $35,000.
+ **GitHub repository:** Primary asset for housing the ActivitySim codebase and communicating about the project. Includes the **GitHub wiki** (see below).
+ **GitHub wiki:** Notable entries to the wiki include: extensive **meeting minutes**, **past**, **current**, and **future development logs**, the project's five **core principles**, and **governance documentation**.

There are additional elements of ActivitySim's governance model detailed across various project documentation — for instance the PMC's voting scheme and guidelines for managing external contributions — that are not captured above. As we'll discuss later, there are a handful of documented governance mechanisms which are not typically put to use, which is why we've omitted them here.

Discussion of governance in ActivitySim

During our research we examined ActivitySim's governance infrastructure through the lenses of design, implementation, and responsiveness that we outlined above. Below, we discuss what we observed.

Governance design in ActivitySim

ActivitySim's approach to designing an initial governance infrastructure included:

+ a centralized design process, and
+ a balance of informal and formal approaches to governance.

Centralized design process

ActivitySim's centralized design process — where one motivated member took on most of the responsibility for drafting an initial governance model, with other members weighing in to refine and finalize — enabled members to agree on and stand up a functioning governance model quickly, and with little friction.

The project's initial governance design defined the project's mission, outlined key roles and responsibilities within the consortium, distinguished roles such as users, developers, committers, and funders, created the Project Management Committee (PMC), and established policies for electing its members. The initial governance documentation also outlined decision-making procedures for a host of potential scenarios including changing code, proposing new committers, selecting PMC leadership, conducting confidential votes, overviewed the project's approach to code management, and indicated how contributors would be recognized by the project.

Governance formality

ActivitySim's initial governance design included both formal and informal governance strategies. Broadly speaking, we refer to formal governance as governance processes which are explicitly understood, written down, and as a result are more easily replicated. Conversely, informal governance refers to governance processes which are implicitly understood, not written down, and as a result tend to be more variable in practice. While ActivitySim's governance documentation architects formal governance solutions for a wide spectrum of potential scenarios, it also acknowledges that the majority of processes won't demand a high degree of formality. For example, the documentation states: "Most day-to-day operations do not require explicit voting — just get on and do the work."

A preference for the informal in day-to-day decision-making creates built-in flexibility for the consortium. As the project always planned to keep membership open, this approach has helped the group avoid becoming beholden to rigid structures and processes as new members join with various perspectives, motivations, and attitudes. But at the same time, the consortium established an explicit framework for more formal governance that serves as an insurance policy against unanticipated challenges and conflict. As one member emphasized:

> "I like things to be informal. I understand the need for governance documentation, and when we started we felt the need to have some type of formal governance in place. But I'll tell you, it wasn't anything time consuming that we had to write and rewrite, or run by our legal department. Yes, you've got to have procedures in place and governance documentation — but I believe in being informal."

Governance implementation in ActivitySim

Notable features of ActivitySim's governance which shape collaboration and development include:

+ a centralized leadership model,
+ several mechanisms and strategies which, together, filter the composition of its membership,
+ a divergence between written and practiced governance, and
+ an informal request for proposal process (i-RFP).

If you ask someone in the consortium, ActivitySim has succeeded in promoting smooth collaboration, with members nearly unanimously expressing how much they enjoy working with each other. As one member commented, "The beauty is that [collaboration] is just happening naturally. It's not like, 'Oh, my God! This meeting again, or you know, 'I hate this meeting,' or 'It is a waste of time.' In my case, I always look forward to it."

While it may feel that way, the consortium's smooth collaboration is far from "natural." Several features of ActivitySim's governance promote productive working relationships between various stakeholders — a central challenge in ISCs as they work to define shared objectives and expectations, align their needs, reconcile divergent working practices, and overcome resource and expertise limitations and asymmetries.

Centralized leadership

ActivitySim's centralized leadership is one mechanism which helps promote clear and productive collaboration. ActivitySim is led by the PMC Chair. The same person has occupied this role since it was created. The current Chair has dedicated significant time and energy to shaping the role's responsibilities and influence on development. As another member put it, "We're very lucky, we have a very strong leader. [He's] super involved, and pushes the entire [project] forward. So his opinion matters a lot."

But as much as its centralized leadership is a boon for the project, it is also a point of vulnerability. Every member of the consortium we interviewed expressed how crucial they think the PMC chair is to the success of the project along with concerns about how reliant they are on the role's current occupant. For example, one member said, "At some point, [the PMC chair] might get a promotion within his agency and it may no longer make sense for him to be the leader. Right now, if we were to lose him, it'd be a very damaging hit, and I'm not one hundred percent sure ActivitySim would come out the same on the other side. It would probably look worse. So how do we protect ourselves against losing our key point person?"

Some members also expressed uncertainty about whether anyone else currently in the consortium is able to shift into the time-intensive, volunteer position of PMC chair, with one person adding, "[The PMC chair] spends so much time on [ActivitySim] alone. I don't know if it is sustainable in the long-term. If [he] can't commit as much time, or if he loses interest, is another person going to step up? It is volunteering your time, you're not paid... if nobody does that, what's the solution moving forward? It is a question mark."

Filtering membership

Another way ActivitySim promotes productive collaboration is by governing membership growth. The consortium's entry requirements, project principles, and open process documentation act in tandem as filtering mechanisms, attracting and retaining members with broadly similar goals, motivations, and attitudes.

Notably, these filtering mechanisms include both light-touch and indirect interventions, such as defined project principles, as well as more heavy-handed and direct ones, such as a mandatory MOA and financial commitment. While more heavy-handed filtering mechanisms may seem more important, lighter-touch ones can be highly influential in governing membership composition. Multiple members reflected on the usefulness of the project's documentation and its clearly defined principles as tools that informed the decision to join the consortium, because they helped illustrate what they were signing up for.

Another notable factor of ActivitySim's governance which functions to support smooth collaboration is the consortium's intentional approach to membership growth. This is an informal mechanism by which current members decide when they are ready to take on more members as well as how much effort they will put into active recruitment. ActivitySim grew relatively quickly to five members when it began, but has seen slower, steadier growth since then. For example, after growing from three to five members, a sixth government approached the consortium with an intent to join. However, at the time, the consortium indicated it wasn't yet ready to take on new members. About a year later, that sixth member did ultimately join the consortium.

This small anecdote reveals that there was a key stage early on when ActivitySim decided to keep its doors closed. However, it has been open and growing — albeit slowly — ever since. Its pace of growth is driven in part by the consortium's decision not to engage in proactive recruitment efforts. However, during interviews we learned that many members do see more active recruitment strategies as an option for growing the consortium's numbers. At least for now members shared that they tend to prefer a more passive approach, as most worry about managing "too many new voices" at once. Strategically using both active and passive strategies for member recruitment enables ActivitySim to maintain an intentional approach to growth even while keeping membership fully open.

Written versus practiced governance

As discussed earlier, the consortium designed a highly formal and forward-thinking governance model soon after its inception. However, due to a preference for more informal day-to-day governance, some members view this model largely— though not entirely — as a back-up plan.

What this means is that implementations of governance are often different in practice from what's written down. For example, the voting scheme outlined by the consortium is rarely, if ever, deployed as detailed in the governance documentation. Another difference: The role of PMC Chair as written touches primarily on managing voting, but in practice is much more expansive. Code testing procedures are not strictly followed, either. For example the written requirement that the PMC establish and maintain unit testing goals isn't strictly enforced.

While this divergence between what's written and what's practiced illustrates the consortium's ability to adapt and remain flexible and responsive to new needs (more on this in the later section on responsiveness), it has at times led to confusion for newer members. Long-standing members have a lived history and implicit understanding of which governance practices diverge from what's written down and why. Newer members, however, who are more likely to have developed expectations of governance by engaging with the written documentation, might be confused by the day-to-day realities of governance.

A newer member spoke to this experience, citing a time they tried to get code developed separately and independently merged into the shared ActivitySim codebase:

> "This is my conjecture, because I've only been in this thing a few years — but it appears that some people who've been around longer know how [updating the code] works, but it doesn't seem to be a written-down, formal process. That [written process] needs to exist because there's a few things I want to do [in the codebase], but I'm not entirely sure how to do them. For example, I've gone off and built a component I want to add [to the codebase], but I'm not sure how to get it added, because there's no document that says 'you have to do all these things and meet these tests'. I mean, there's a procedure, but it is all in people's heads."

Internal vendor market

Another tool ActivitySim wields to promote is its informal request for proposal (i-RFP) process. The i-RFP process asks an internal market of bench vendors to submit proposals in response to the PMC's annually-scoped priorities. This is a creative strategy that offers the consortium the opportunity to capitalize on the oft-cited benefits of competitive bidding processes in government, while maintaining the stability and institutional expertise that working continuously with the same vendors can offer.

Governance responsiveness in ActivitySim

ActivitySim's governance affects the consortium's ability to anticipate and navigate future trade-offs, both known and unknown, via:

+ the consortium's view of its governance infrastructure as dynamic and adaptable, and
+ its frequently held meetings on development.

Dynamic view of governance

The ActivitySim consortium's dynamic stance toward governance means governance infrastructure can be adapted in response to emerging needs. One example of this dynamic approach is the consortium's recent vote to add a vice chair to the PMC in response to the concerns about its highly centralized leadership model. One member's reflection is illustrative of this general preference for an iterative, agile approach to governance:

> "The way the project was originally set up, there was a single lead. But the coordination just got to the point where that no longer made sense. And the thought in my head is: Had we started with shared leads, would that group have been productive? Or would they have had nothing to do and thought, 'Well, my position isn't really needed,' and gotten into a pattern that wasn't helpful? But we started with a single point and saw how that person got stretched and what needed to be done. So I think we might be in a better position to adapt because of that."

Another exmaple is the consortium's adaptation of the governance mechanisms which AMPO stewards. like changing the format of the memoranda of agreement which made the process of joining and leaving the consortium clearer and easier. Even more broadly, the consortium views its core relationship with AMPO as adaptable. As a result, AMPO's role has significantly changed over time, expanding to support core project management and promoting the retention of organizational knowledge even if particular members change. As one member explained:

> "AMPO's role itself has evolved from really just being a pretty neutral vehicle through which contracts could be established into having a significant role. [I]t's been good to have them play a more active role in the overall project management as a neutral party that persists when individual agencies might join or or cease to participate."

Further illustrating the consortium's dynamic stance toward governance: Some members anticipate that their needs for AMPO's role will continue to shift. This led one member to ask whether "AMPO is the mechanism that will keep [ActivitySim] functioning long-term" and if such a small organization has adequate "legal and liability protection, the staffing, and the dollars" to meet the consortium's future needs as it grows.

Internal vendor market

The consortium's frequent meetings promote governance responsiveness by minimizing the opportunity for issues to become severe. In addition, the consortium's twice weekly meeting with vendors functions as a vital communication channel between the PMC and vendors. During this time, both the PMC members and vendors can ask and address granular questions about development and user needs, agile development is promoted, and plans for review and testing are solidified.

Notably, the vast majority of meetings are focused on development-related topics. There are fewer opportunities for members to collectively reflect on and adapt governance infrastructure, despite their shared view of governance as dynamic. This means ActivitySim's governance is not necessarily as dynamic as it could be — or as members might hope it to be.

While there are certainly ways in which the consortium's dynamic approach to governance has manifested in practice, as discussed above, some members hold concerns about potential issues on the horizon" — such as future membership growth or increases in the use of ActivitySim as the primary production model across member governments. These issues can benefit from what one member referred to as "governance tune-ups". But the consortium can struggle to find dedicated time to "tune up", or adapt, its governance in anticipation of these longer-term needs.

As one of ActivitySim's developers shared: "Updating governance is a key job of the consortium, but people often don't have time or don't want to prioritize it because there are always shinier things you could spend your time and money on." The result: changes and updates to ActivitySim's governance are largely reactive, rather than proactive.

In addition to the consortium's frequent touch points being largely limited to development-related topics, these meetings also tend to center on granular, short-term development goals, rather than long-term product vision. This means that the consortium has few opportunities to govern long-term development. At present, active management of a high-level product vision is a gap in its governance infrastructure. This gap leads to some uncertainty among members, who wonder whether the vision for ActivitySim is a component library, a fully-fledged plug-and-play product, or something in between. While members generally feel confident and satisfied with the project's short-term development trajectory, they sometimes struggle with feeling similarly confident in making clear, informed decisions about which direction to orient the project in the long-term — or even in understanding what's possible. Explained one founding member:

> "At the very beginning we had this dream to create a software platform that was just plug in and plug out, you know? That just works for everyone. We quickly realized with the complexity of behavior modeling, it seems like that is not going to happen. Unless some super capable software engineer can design a system that reforms the way it works right now. But the way it is right now... probably not."

These issues pop up because members and vendors may have different software engineering expertise. While most meetings focus on granular issues of development, vendors usually have more development expertise than members. This is a challenge commonly seen in government software projects. As a result, many members don't feel comfortable speaking up to influence decisions that impact long-term product vision. Instead, they rely heavily on vendors to make big-picture, long-term decisions about product development and architecture, as one member points out:

> "Many of us have a computer science background and can program, but to understand software engineering and architecture... that is a totally different level. I don't think I'm in the right position to provide guidance on what [the engineers] should do. [During meetings] we talk about faster runtimes, better memory management, modularity, future maintenance costs... but how much can I contribute to that conversation? That is my question. I think that is a common feeling among all the consortium members."

Architecting effective governance in ISCs

Challenges to productive collaboration and software development are inevitable in ISCs. One might argue that the ideal goal for an ISC is not to eliminate issues or conflict, but to create a governance infrastructure which allows it to quickly and effectively respond to both anticipated and unanticipated issues. By studying ActivitySim we hoped to uncover and share practical, actionable lessons to help other ISCs establish more effective governance that helps them achieve this goal. Examining ActivitySim's governance in action revealed several lessons about:

+ The roles of adaptability and formality in establishing effective governance
+ How governance impacts ISC membership composition
+ Governance strategies for promoting leadership sustainability and empowering government staff in vendor relationships
+ Governing meeting cadence and topics to avoid severe conflict.

Below, we lay out the lessons that emerged from this case study and discuss practical recommendations to help practitioners translate them into practice.

Six lessons in ISC governance

Lesson 1: A dynamic approach to governance helps ISCs adaptively meet challenges as they mature

A dynamic approach to governance acknowledges that unanticipated changes and challenges are likely to crop up. This strategy also acknowledges that a first pass at governance design is unlikely to perfectly meet an ISC's needs as it grows and matures. Operationalizing a dynamic view of governance helps ISCs avoid feeling too rigid about a first pass at governance design because everyone understands that new needs can be addressed down the line.

To ensure governance can be dynamic and adaptable in practice and not just in theory, ISCs should ensure that initial governance design includes clear mechanisms for proposing, assessing, and implementing new governance structures. This could be, for example, a short template members complete to propose a new governance process and a simple voting system.

ISCs should also dedicate time to discuss and refine governance infrastructure separately from spaces where development is discussed to ensure governance needs are appropriately prioritized, and don't become an afterthought.

Finally, running regular retrospectives in governance-focused spaces can be a useful way to proactively discover new governance needs as they arise.

Lesson 2: A balance of formal and informal governance strategies promotes clarity and helps ISCs respond to diverse challenges

This lesson is a clarifying update to our prior advice which at times suggested that governance doesn't need to be formal. Indeed, informal governance mechanisms can enable ISCs to retain flexibility. That said, even if an ISC prefers to govern day-to-day work informally, successful and sustainable ISCs have some level of formal, documented governance structures in place to promote shared understanding and to insure against inevitable conflict.

Foundation for Public Code designed a tool called the Governance Game that ISCs should consider playing. The game helps groups collaborating on public code architect governance, both formal and informal, that is equipped to handle (un)expected challenges.

Lesson 3: Intentionally governing member growth is more important than starting small

The Beeck Center's previous advice on starting an ISC emphasized the importance of starting small with two members to avoid a "too many cooks" scenario. But through our continued research and engagement with the ISC community, we've now come across many successful ISCs that started with more than two initial members.

Considering this fact through the lens of governance, it is now clear that intentionally governed growth is more important than starting small. While starting with two members isn't necessarily bad advice, it is too conservative and may needlessly limit growth. For instance, one potential downside to starting too small is that development may be overly tailored to a limited set of needs early on. This could make expansion to a broader set of shared needs more difficult down the line.

While being open or closed to new members is one approach to achieving this intentional governance, an ISC that is consistently open to new members can maintain that intentionality, for example by calibrating how actively it recruits new members. A key component of intentional governance of membership growth, then, is not only deciding whether or when an ISC will be open to new membership, but how. ISCs can explore strategies for how members are recruited, how criteria for membership eligibility are designed and implemented, and how new members are onboarded.

Additionally, governance mechanisms affect how potential members view an ISC and can "filter" membership, attracting and retaining members with particular kinds of goals, motivations, and attitudes. Governance mechanisms with obvious filtering effects are heavy-handed ones, such as contracts or financial commitments. However, it can be easy to overlook how lighter-touch mechanisms — such as defining project principles — can also impact the composition of an ISC's membership base.

One way to prevent overlooking governance mechanisms that impact membership is by engaging potential members, not just current ones, during governance design. This can help ISCs understand whether and how external perceptions of governance — existing or proposed — affect external organizations' decisions to join. This user research can then inform how particular governance mechanisms are designed or updated.

Lesson 4: Distributing leadership responsibilities and planning for succession promotes sustainable and resilient leadership

Previous Beeck Center research suggested that having a single, visionary project leader is an exemplary model for other ISCs, but this can be difficult to replicate. While we maintain that it is important to have a single product owner steward a vision, in an age where government staff tend to experience burnout and turnover at relatively high rates, overly centralized leadership doesn't promote resilience unless accompanied by additional strategies which help distribute responsibilities and support succession planning.

Strategies that promote sustainable and resilient leadership for ISCs could include creating new leadership positions, mandating turn-taking in leadership service as a requirement of membership, or offering discounts on membership dues for organizations that allocate staff time for leadership roles.

Lesson 5: Governance can help ISC members remain empowered in vendor relationships

Government staff often feel unable to effectively manage software vendors or question their approaches because they feel they lack the requisite expertise. ISCs are not exempt from this common challenge.

Governance strategies which promote competitive, outcome-driven contracting and which work to minimize knowledge gaps between government staff and software developers promote empowered and effective government-vendor relationships in ISCs. Helpfully, there's abundant and extensive guidance available on these topics.

Creative governance of procurement, such as ActivitySim's i-RFP approach, can help maintain competition in contracting even when working with bench vendors. Agile and modular procurement strategies serve to de-risk vendor relationships and tie procurement contracts to services that drive toward outcomes. These strategies can make managing vendor contracts easier and more likely to lead to performant software solutions.

There is also a suite of development practices that can support a governance model aimed at bridging knowledge gaps between government staff and external engineers. Developing user-centered Objectives and Key Results (commonly known as OKRs), having a product manager steward a vision and roadmap for the software, and ensuring vendors implement ADRs are all useful and practical strategies. Each one helps non-developers better understand, engage with, and manage vendors and the development process.

Lesson 6: Frequent touchpoints mitigate the potential for severe conflict

ISCs are composed of many governments with different and sometimes competing needs. Frequent touch points allow issues and concerns of all types to be resolved at early stages. This strategy helps prevent bigger, more severe challenges and conflicts from arising.

Exactly what frequent means will vary for individual ISCs, but generally speaking any members involved in governance should meet at minimum once per week. As discussed above, ISCs should dedicate some of these touch points to topics besides short- and mid-term development — for example governance and product vision — to ensure all issues are aired and addressed.

Limitations

When considering the high rate of positive, collaborative experiences with collaboration that members in the ActivitySim consortium, we must keep two factors in mind. First, the government TDM community is small. This means the stakes are high for maintaining good relationships. Second, TDM is a highly professionalized and codified field, which means that variance in core systems and processes, skills and competencies, and attitudes and goals across organizations may be limited. Together, these unique factors minimize opportunities for conflict — explaining at least in part the near-frictionless collaboration within the ActivitySim consortium — and may not be highly replicable.

It should also be noted that ActivitySim is an open-source consortium, which is a relatively direct model of governing an ISC. Government members sit on the ISC's governing body and play a deep and active role in governance. While we do think the lessons uncovered through this case study are likely to apply to other models, there may be additional lessons to glean from studying different models. For example, ISCs with more representative governance — such as those with a governing body not made up of its government members or ISCs governing closed-source solutions — may have different governance needs, experiences, and outcomes.

Conclusion

In our seminal landscape research on software sharing and collaboration in government, we identified governance as a key factor behind the success or failure of ISCs. This research builds and iterates on our previous work by offering six actionable lessons on achieving clear, effective governance in ISCs.

What remains clear is that governance is indeed a crucial determinant of an ISC's success. Government practitioners who work in ISCs and dedicate the time, energy, and resources to governance that it deserves will be well-positioned to foster smooth collaborations that produce effective shared software solutions.

beeckcenter
social impact + innovation

BEECK CENTER FOR SOCIAL IMPACT + INNOVATION 37th and O Streets, NW, Washington DC 20057 | beeckcenter@georgetown.edu