



Interoperabilità e standard nei sistemi informatici della PA: il loro valore per lo sviluppo di una PA digitale e di un mercato aperto e di qualità.

Introduzione.....	1
Alcune definizioni	2
Sistema	2
Interoperabilità	3
API (Application Programming Interface).....	3
Interoperabilità e standard	4
I vantaggi di un approccio "sistemico".....	5
La declinazione del concetto di interoperabilità per la PA	6
Lo sviluppo dei sistemi informatici dei Comuni	8
Correre ai ripari	11

Introduzione

Giustamente indicata come uno degli elementi chiave per la realizzazione dei sistemi informatici delle PA, l'interoperabilità è diventata uno degli argomenti di cui maggiormente si parla all'interno della Pubblica Amministrazione.

Come spesso però accade quando si parla di concetti alla moda, si rischia di semplificare eccessivamente, a volte addirittura di parlare di concetti senza nemmeno averli definiti, non almeno con la necessaria precisione. Finendo così per creare confusione e veicolare messaggi non propriamente corretti.

Proviamo a fare un po' di chiarezza, cercando di capire cosa realmente è l'interoperabilità, specie in un contesto come quello della PA (in particolare quello comunale), caratterizzato dal dover supportare una grande quantità e varietà di funzioni in capo agli Enti, una normativa assai prescrittiva (e invasiva) nonché scarsi spazi di manovra a livello organizzativo. Condizioni queste assai sfidanti, dove l'interoperabilità, l'integrazione e gli standard, svolgerebbero realmente un ruolo decisivo nella realizzazione del sistema informatico.

Alcune definizioni

Sistema

Parliamo spesso di "sistema" informatico, ma saremo in grado di spiegare cosa intendiamo con questo termine? A mio parere la definizione che meglio si adatta, anche nell'ambito informatico è quella data da Galileo:

«Pluralità di elementi materiali coordinati tra loro in modo da formare un complesso organico soggetto a date regole»

La definizione mette bene in evidenza quali sono le caratteristiche che compongono ciò che usualmente viene indicato come sistema:

- ci devono essere più componenti, altrimenti tutto quello che segue non ha senso (è ovvio che un solo componente è coordinato con sé stesso);
- devono lavorare in modo organico e coordinato, per giungere allo scopo a cui il sistema è destinato;
- il coordinamento delle risorse è basato su regole ben precise, spesso sotto forma di **standard**.

Vedremo in seguito come, in base a questa definizione di sistema, molti sistemi informatici nelle PA, almeno per quanto concerne i software, veri e propri sistemi non sono.

Interoperabilità

La definizione di interoperabilità "classica" è quella definita dalla norma ISO/IEC 2382:

*«The capability to communicate, execute programs, or transfer data among various **functional units** in a manner that requires the user to have **little or no knowledge** of the unique characteristics of those units»*

La definizione, necessariamente generica, ma specifica dell'IT, mette in evidenza anche in questo caso le caratteristiche fondamentali del concetto definito:

- la capacità di comunicare o eseguire operazioni fra diverse unità funzionali (che per il nostro scopo possiamo identificare con le parti del sistema)
- il tutto senza che non sia necessario conoscere i dettagli implementativi delle singole unità funzionali (caratteristica chiamata "blackboxing")

Si può notare che, di fatto, per definizione, un sistema informatico, per potersi definire tale, è per sua stessa natura interoperabile.

Da notare inoltre come il concetto di interoperabilità sia in grado di coprire casistiche assai diverse, anche relative a sistemi diversi, a parti omogenee di sistemi diversi, ecc.

Quando si parla di interoperabilità è quindi necessario anche specificare fra chi/cosa, ovvero la declinazione che, nel contesto specifico, si vuole dare all'interoperabilità. Vedremo, ad esempio, come nel Piano Triennale questa sia declinata in modo assai specifico e lasci scoperti numerosi ambiti, la cui mancanza genera evidenti problemi e che, pertanto, andrebbero in qualche modo coperti.

API (Application Programming Interface)

Altro acronimo diffusissimo. Ma cosa c'entrano con sistema e interoperabilità?

Le API sono l'implementazione "tecnica" dell'interoperabilità fra i software. Affinché due unità funzionali (software, moduli software, ecc.) possano fra loro interagire è necessario che questi mettano a disposizione delle apposite interfacce di programmazione, con cui realizzare questa comunicazione.

Attraverso le API i software mettono a disposizione a tutti gli altri software le loro "capacità" (o dati/documenti). Così se un software deve realizzare un'operazione che già un altro software sa fare, invece che implementarla ex novo internamente, semplicemente sfrutta la capacità dell'altro software. Con evidenti vantaggi.

Il concetto di API non è nuovo, anzi, è presente da almeno 40 anni. Tuttavia, solo più recentemente grazie all'introduzione di tecnologie basate sugli **standard del web** (http, html, xml, json, ecc) le API hanno assunto un'importanza ed una diffusione rilevante. Chi non ha almeno una volta sentito parlare di SOAP o ReST?

Da notare come anche in questo caso, è stata l'introduzione di uno standard a far esplodere, non a caso, il fenomeno.

Interoperabilità e standard

Una nota pubblicità di pneumatici aveva uno slogan assai efficace per rimarcare l'importanza del prodotto: "la potenza non è nulla senza controllo". Ebbene, parafrasando questo slogan si potrebbe affermare che "l'interoperabilità non è nulla senza gli standard".

Concetto che dovrebbe risultare evidente ma su cui è bene soffermarsi. Come si diceva per le API, il fenomeno è esploso nel momento in cui per implementare le API è stato definito uno standard.

Prima ognuno che implementava un'API si "inventava" un suo modo il che rendeva complicata l'implementazione ma soprattutto l'utilizzo, specie per chi aveva la necessità di comunicare con API diverse di produttori diversi. Per ognuna di queste doveva implementare una tecnologia diversa. Spesso queste tecnologie erano assai complesse e costose da implementare¹. Poi con l'avvento e l'enorme diffusione degli standard per il web si pensò di definire un modo di invocare le API basato su questi. Nacque dapprima lo standard SOAP e, successivamente, quello che attualmente è lo standard di fatto, il ReST.

In questo modo risultò molto più semplice sia esporre funzionalità che fruirne. Ma soprattutto ogni API poteva essere implementata e utilizzata in un unico modo. Da quel momento il numero di API disponibile aumentò esponenzialmente.

Un primo passo "tecnico", fondamentale, ma non ancora completo.

Passo, questo, che è sufficiente solo in particolari contesti caratterizzati dall'aver un solo software a cui fare riferimento per certe funzionalità.

È il caso, nella PA, di erogatori unici, come possono essere INPS, INAIL, AdE, ecc. In casi come questi è il singolo erogatore che detta gli standard, gli altri si adeguano.

Ma se così non è, e gli erogatori sono molteplici, oltre all'interoperabilità "tecnica" serve altro.

Supponiamo, ad esempio, che io acquisti un sistema contabile e che questo esponga delle API ReST in modo che altri software utilizzino le sue funzionalità,

¹ È il periodo in cui imperarono i cosiddetti "middleware", software che altro non facevano se non costruire un "ponte" più agevole per interagire con un'API complessa.

ad esempio la capacità di registrare contabilmente una fattura. I miei software altro non dovranno fare che utilizzare i servizi ReST messi da questo a disposizione. Non mi servirà altro che "integrare" i miei software attraverso queste API e il gioco è fatto.

Ma se volessi cambiare software contabile, cosa succederebbe? Il nuovo software contabile esporrà delle API, ma molto probabilmente, la loro interfaccia sarà diversa, potenzialmente molto diversa. Risultato: dovrò riaggiornare, se non addirittura re-implementare, tutta la parte del software che utilizza le API. Cosa spesso complessa ed onerosa, che vincola nel cambio del software.

In questo caso la presenza di standard è fondamentale per rendere l'interoperabilità, permettetemi il gioco di parole, interoperabile.

Sarebbe qui necessario fare di più²:

- definire uno standard per la "segnatura" delle API (funzioni uguali definiscono API uguali)
- definire uno standard per la codifica dei dati (ecco qui, un esempio dove un progetto come padocs.it trova la sua strategicità)³

In questo modo il cambio di un software non produrrebbe problematiche di portabilità, perché ogni altro software che vi si integrava, troverebbe, nel suo sostituto, la medesima interfaccia e codifica dei dati. In sostanza in caso di cambiamento del software, gli altri software altro non dovrebbero fare che, al più, aggiornare l'end-point (ovvero l'"indirizzo" a cui trovare il nuovo software).

I vantaggi di un approccio "sistemico"

I vantaggi di un approccio all'interoperabilità così esteso sono numerosi ed evidenti:

- costituzione di un sistema realmente modulare, con singoli moduli/componenti che possono essere sostituiti in modo semplice e non oneroso;
- si modera il rischio di vendor lock-in, in quanto la necessità di dotarsi di componenti dello stesso fornitore viene meno;
- si amplifica, di conseguenza, il potere contrattuale per chi acquista;
- si stimola la specializzazione, che a sua volta innalza la qualità dei prodotti e si apre il mercato, altrimenti appannaggio di solo grandissime aziende;
- la definizione di standard anche a livello di codifiche dei dati, agevola il cambio dei software, con un doppio vantaggio: per gli Enti, che possono ridurre i costi di migrazione dei dati in caso di cambio del software, di

² La possibilità di operare in questo modo non è legata alla tecnologia ReST, si può comunque realizzare; ad esempio nel mondo delle firme digitali, ci si astraie dai differenti chip presenti sul mercato utilizzando uno standard e implementandolo sotto forma di libreria binaria standardizzata (dll nel mondo Windows – in genere denominata *pki.dll)

³ L'esempio della fattura non è casuale, è infatti grazie allo standard introdotto per la fatturazione elettronica che le fatture, o meglio, i dati in esse contenuti, sono "portabili".

mercato, che diventa più aperto e realmente concorrenziale; la standardizzazione delle codifiche, infine, agevola l'interoperabilità anche in un contesto come la PDND, dove dati uguali possono più facilmente essere esposti in modo uguale da tutti gli erogatori.

Ma i vantaggi non si limitano solo a questo, perché sono evidenti anche per i produttori dei software, specie se assai complessi, in quanto la modularità, la segmentazione in componenti più piccole e autonome (il cosiddetto disaccoppiamento)⁴ rendono la realizzazione ma soprattutto la manutenzione alquanto più agevole. Un aspetto questo, in un mondo assai mutabile come quello della PA, non irrilevante.

È evidente che un approccio simile non è sempre raggiungibile, ma in contesti ben delineati, come appunto quello della PA, dotata fra l'altro di enorme potere contrattuale, non solo è realizzabile, ma auspicabile e si può imporre anche a livello normativo o regolamentare, magari ponendolo come requisito per la qualificazione dei prodotti.

Perché quindi non si fa?

La declinazione del concetto di interoperabilità per la PA

Alla base di tutto c'è il modo in cui è declinato il concetto di interoperabilità nella nostra normativa.

Nel CAD, ad esempio, si definisce come: *"la parte del Sistema Pubblico di Connettività finalizzata all'interazione tra i sistemi informatici dei soggetti partecipanti, per garantire l'integrazione dei metadati, delle informazioni, dei processi e procedimenti amministrativi"*

Nel Piano Triennale: *"un asse portante dell'intero sistema informativo pubblico: assicura l'interazione e lo scambio di informazioni tra le PA senza necessità di specifiche integrazioni, garantendo la piena collaborazione tra le amministrazioni pubbliche e i soggetti privati"*

Questa definizione di interoperabilità è incentrata sulla capacità di comunicare fra sistemi di PA diversi, piuttosto che fra componenti di un sistema. In altre parole, ad esempio, si cerca di far parlare il protocollo dell'Ente "A" con quello dell'Ente "B" oppure, caso ancora più tipico, consentire, in presenza di un sistema erogatore "centrale", ad altri sistemi, di accedere ai dati/funzionalità di

⁴ Attualmente l'architettura del software più alla moda è proprio quella detta a "microservizi" che appunto, si basa sul concetto di segmentare il software in elementi funzionali minimali e disaccoppiati.

questo. L'ultimo caso non presuppone, fra l'altro, nemmeno la definizione di standard, ma è il sistema centrale stesso a dettarli.

La stessa PDND e le relative regole tecniche sull'interoperabilità⁵ trattano principalmente di "standard tecnici" regolando le modalità di implementazione delle API e, nel caso della PDND, standardizzando autenticazione/autorizzazione.

In realtà le succitate Linee Guida precisano come anche la "semantica" sia da standardizzare. Nel Paragrafo 4.3 si troverebbe un passaggio fondamentale:

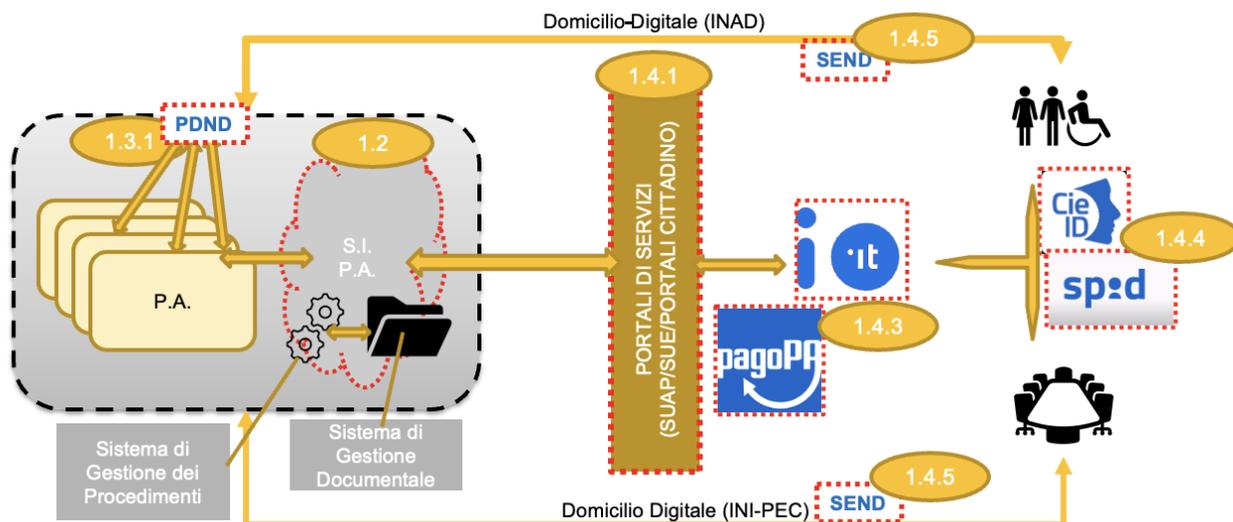
*"La comunicazione tra soggetti **DEVE utilizzare modelli dati condivisi**, in modo da razionalizzare e uniformare la rappresentazione dell'informazione quale presupposto per favorire l'interoperabilità **tra soggetti differenti**. L'adozione di modelli dati condivisi permette ai soggetti che comunicano, di partire da una rappresentazione dei dati comune, evitando la proliferazione di modelli dati differenti impattano direttamente sulla modalità con cui i dati vengono resi persistenti. Ne consegue che l'adozione di modelli dati condivisi favorisce il riuso delle strutture dati, semplifica le integrazioni e assicura che gli elementi dei dati siano compresi nello stesso modo tra le parti tra loro comunicanti".*

Che è esattamente quello che si diceva sopra. Il problema è che questa indicazione è rimasta di fatto lettera morta⁶ e in ogni caso si concentra solo sull'interoperabilità fra Enti, non prendendo minimamente in considerazione la definizione di standard per l'interoperabilità fra i diversi moduli del singolo sistema. Anche in tema di interoperabilità fra Enti gli standard latitano, e la dimostrazione eclatante di questo si vede nei risultati prodotti dall'Avviso 1.3.1 di PaDigitale2026. Qui i Comuni sono stati chiamati ad esporre attraverso la PDND delle API, ma senza a monte definire uno standard. Così ogni Comune può esporre il medesimo dato del Comune vicino con API e codifiche dei dati completamente incompatibili. Assolutamente inutile!

Non solo, la stessa strutturazione degli Avvisi di PaDigitale2026, mostra evidenti segni di questa impostazione fortemente segmentata, che non aiuta ad inserire i vari progetti in un disegno complessivo, che invece esisterebbe.

⁵ Vedasi [Determinazione Agid n. 547 del 1 ottobre 2021](#) - Adozione delle "Linee guida Tecnologie e standard per la sicurezza dell'interoperabilità tramite API dei sistemi informatici" e delle "Linee guida sull'interoperabilità tecnica delle Pubbliche Amministrazioni" e [Determinazione Agid n. 627/2021 del 15 dicembre 2021](#) - Adozione delle "Linee Guida sull'infrastruttura tecnologica della Piattaforma Digitale Nazionale Dati per l'interoperabilità dei sistemi informativi e delle basi di dati" ai sensi dell'articolo 50-ter, comma 2 del D. Lgs. 7 marzo 2005, n. 82 e s.m.i.

⁶ Questi standard si dovrebbero trovare nel [Catalogo Nazionale per lo scambio dei dati e le informazioni fra le PA](#) che però si trova in uno stato "embrionale"



Gli Avvisi di PaDigitale2026 rappresentati nel loro disegno complessivo

Non solo, nei singoli Avvisi mai una sola volta è esplicitamente inserita una clausola di integrazione nel contesto del sistema informatico dell'Ente attuatore. Così vediamo per servizi online che non comunicano con i backoffice, piattaforme per la notifica di atti che non interagiscono con i sistemi di gestione documentale, servizi di pagamento che non interagiscono con i sistemi contabili o quelli che hanno originato le posizioni debitorie.

Tutto questo ha delle importanti ricadute sulle soluzioni presenti sul mercato per lo sviluppo dei sistemi informatici degli Enti, specie dei Comuni.

Lo sviluppo dei sistemi informatici dei Comuni

Il caso dei Comuni è emblematico, per diversi motivi. In primo luogo, per via delle caratteristiche peculiari dell'Ente comunale che si contraddistingue per:

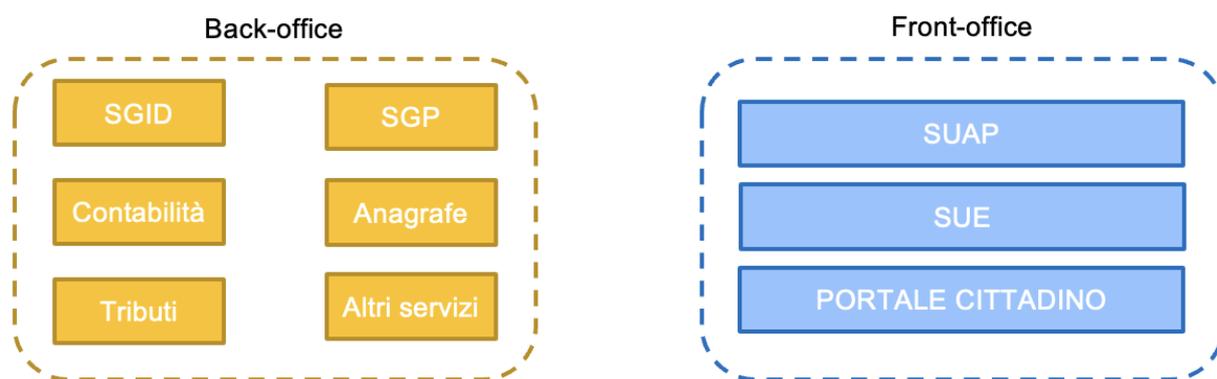
- essere un Ente di "prima linea", uno dei punti di contatto privilegiati per l'utenza;
- essere uno degli Enti a cui sono delegate il maggior numero di funzioni⁷
- essere un insieme di Enti molto diversi fra loro: si va dal micro Comune con meno di dieci dipendenti al grande capoluogo con migliaia di dipendenti, magari a loro volta, come succede a Roma, spezzati in diversi "municipi".

Queste caratteristiche fanno sì che la costruzione di un sistema informatico di un Comune sia tutt'altro che banale perché si compone di numerosissimi "moduli" (contabilità, anagrafe, stato civile, tributi, protocollo, bilancio,...) che

⁷ Per convincersene basta osservare l'articolazione del Piano di Classificazione dei Comuni, che ne descrive le funzioni: è forse il più articolato fra quello delle PA

devono comunque comunicare fra loro, ed essere in grado anche di esporre il tutto tramite i relativi servizi online.

L'evoluzione di questi sistemi ha principalmente seguito il trend che storicamente ha caratterizzato lo sviluppo dei sistemi nella PA. Si è partiti dal classico modello a "silos" dove ogni ambito è stato informatizzato in modo totalmente separato con software totalmente isolati. Lo scambio di dati avveniva, nella migliore delle ipotesi, attraverso esportazione/importazione di file.



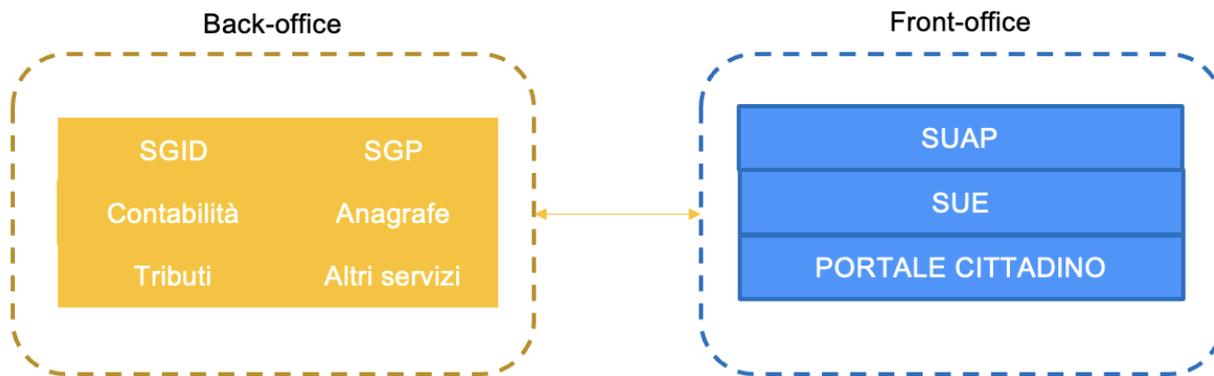
Tipica situazione a «silos», ogni componente del sistema è isolato e replica al suo interno funzionalità di altre componenti (es. un suo repository per i documenti, un suo workflow, ecc, **ci sono tanti sistemi**)

Il limite di un siffatto sistema informatico è evidente e non appena le tecnologie hanno consentito di rendere i vari software capaci di comunicare, c'è stato un tentativo di rendere i diversi moduli software fra loro interoperabili.

Tuttavia, **la mancanza di standard**, ma anche il consolidamento di alcune prassi erranee che il modello a silos aveva introdotto⁸, ha reso questa operazione alquanto difficoltosa, complessa (e onerosa) alla portata di solo pochi grandi Enti.

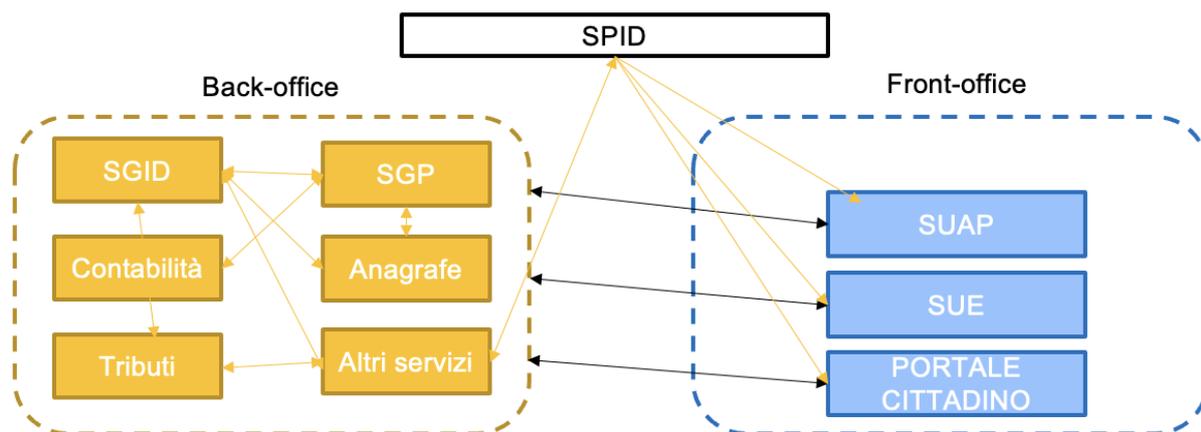
A questo punto il mercato, per soddisfare le esigenze di "integrazione" manifestate dai Comuni, ha trovato più comodo (e redditizio) proporre delle soluzioni "monolitiche", in sostanza un unico grande software, che copre tutte le funzioni del Comune. Il monolite, va da sé, è "integrato" (con sé stesso, *sa va sans dire*) e promette di risolvere tutti i problemi di integrazione. Promessa che purtroppo non è spesso nemmeno realmente mantenuta visto che questi software sono quasi sempre la cucitura dei vecchi moduli pensati a silos.

⁸ Ad esempio, la diffusissima, dannosissima e tutt'ora difficile da eradicare pratica, che vede ogni software proprietario dei documenti che lui produce o gestisce.



Il sistema «monolitico» è in realtà una sola applicazione che fa tutto; è «integrata» in quanto unica, viene spesso proposta come soluzione in grado di creare un sistema. In realtà è fortemente sconsigliabile perché introduce notevoli problematiche (vendor lock-in, mancanza di scalabilità, difficoltà di manutenzione, evoluzione, aggiornamento, ecc.)

È più che evidente che l'evoluzione verso questo tipo di soluzioni, rappresenta una sorta di **antitesi** rispetto alla costruzione del sistema interoperabile che avevamo descritto in precedenza e che si potrebbe rappresentare con la figura qui sotto:



Un sistema è composto da componenti funzionalmente ben definite e distinte, ognuna svolge il suo compito e mette a disposizione delle altre, le funzionalità che implementa, che a loro volta le utilizzano quando necessario. Componenti esterne (qui SPID) fungono da componenti funzionali comuni a vari Enti, realizzano il «sistema della PA»

Ma è altrettanto evidente che fintanto l'interoperabilità continuerà ad essere declinata come nel Piano Triennale molto difficilmente si potrà mirare alla costituzione di un siffatto sistema, rendendo quella della soluzione monolitica l'unica percorribile.

Soluzione monolitica che per altro, se costruita rispettando le attuali linee guida sull'interoperabilità rimane comunque lecita e accettata, portando con se tutte le controindicazioni del caso che tanto hanno limitato anche l'evoluzione dei sistemi informatici dei Comuni e gli stessi progetti finanziati attraverso gli Avvisi di PaDigitale2026.

Correre ai ripari

Come fare quindi per sbloccare questa situazione, non certamente ottimale?

Credo che sarebbe necessario intervenire con una serie di interventi mirati a predisporre un'ambiente sia normativo che tecnico, in grado di stimolare la giusta evoluzione dei sistemi informatici comunali.

Solo per citarne alcuni:

- estendere nel Piano Triennale il concetto di interoperabilità in modo da poter ricomprenderne anche i casi di "interoperabilità interna" superando l'attuale definizione che vede l'interoperabilità solo fra componenti omogenei di sistemi informatici di PA diverse;
- rafforzare la parte "semantica" dell'interoperabilità, magari dedicando maggiori risorse e priorità alla definizione di standard sia per le interfacce sia per le codifiche dei dati, in modo che dati uguali siano esposti con interfacce e codifiche uguali, specie in ambiti, come ad esempio quello comunale, dove i soggetti interoperanti sono molti; il luogo (schema.gov.it) ci sarebbe ma andrebbe profondamente rivisto nell'ottica dell'estensione di cui al punto precedente;
- imporre questi standard a chi vuole entrare a far parte del marketplace di ACN, vincolando la qualificazione al loro rigoroso rispetto;
- ultimo, ma non per questo meno importante, formare le figure negli enti addette all'acquisizione dei software e ai progetti di trasformazione digitale⁹, in modo da renderle consapevoli dei vantaggi che un sistema realmente interoperabile e basato su standard può portare all'Ente.

⁹ RTD, ma anche Eesponsabile per la Gestione Documentale e le varie figure specialistiche che la reente normativa ha introdotto.