

L^AT_EX nella pubblica amministrazione. La web application FAC^ILE per la produzione automatica di comunicazioni interne del Comune di Napoli

Agostino De Marco, Paolo Eugenio Cresci

Sommario

In questo articolo viene presentata la web application FAC^ILE in uso presso l'Amministrazione Comunale di Napoli. L'applicazione è stata rilasciata nel marzo 2011. Essa è usabile attraverso la intranet del Comune di Napoli ed è attualmente in fase di sperimentazione. Gli utenti sono in grado di produrre comunicazioni ufficiali ad uso interno secondo le specifiche del manuale della *Corporate Identity* del Comune di Napoli. L'applicazione è pensata per essere di semplice utilizzo e presenta all'utente una *web form* opportunamente progettata per la produzione di una lettera in formato PDF. Dietro le quinte FAC^ILE utilizza il motore di composizione X_YL^AT_EX ed un *parser* sviluppato in linguaggio PHP. L'aspetto chiave è la progettazione di un *template* di sorgente L^AT_EX basato sulla classe `scrlltr2` appartenente al *bundle* KOMA-script.

Abstract

The article introduces the web application FAC^ILE, in use at the Administration of the City of Naples. The application was released in March 2011. It is integrated into the intranet software tools of the Municipality and is currently under testing. Users are able to produce official internal communication letters which comply to the City of Naples Corporate Identity specifications. The application is designed to be perceived as a very simple tool and presents itself to the user as a web form for the production of a PDF document. Behind the scenes FAC^ILE runs the X_YL^AT_EX typesetting engine together with an *ad hoc* parser developed in the PHP language. The key aspect of the application is the design of a template of L^AT_EX source based on the `scrlltr2` document class from the KOMA-script bundle.

1 Introduzione

Negli ultimi anni la necessità di abbattimento dei costi aziendali, tanto nel settore pubblico quanto in quello privato, ha fatto emergere un crescente interesse verso il mondo del software libero e *open source*. L'Amministrazione comunale di Napoli è un esempio di ente pubblico che ha saputo mostra-

re buone intuizioni nel campo dell'ottimizzazione dei sistemi informativi. Alla fine del 2007, con la delibera n. 3999 quale atto di indirizzo in tema di sviluppo tecnologico e centralizzazione degli acquisti, la Giunta comunale di Napoli indica le modalità per la graduale migrazione verso l'uso di tecnologie a codice aperto: (i) azioni pilota che dimostrassero l'applicabilità del percorso di sviluppo; (ii) priorità nella predisposizione e partecipazione a progetti cofinanziati con fondi europei, nazionali e regionali previsti per la diffusione dell'uso del codice aperto; (iii) specifici e differenziati percorsi formativi del personale, nell'ambito della formazione annualmente erogata dall'Ente. Nel 2009 gli organi dirigenti dell'Ente hanno disposto la formazione di un Gruppo di lavoro interservizi denominato "Ricambio del parco microinformatico del Comune di Napoli e passaggio generalizzato all'*open source*". Questa iniziativa ha portato al rinnovamento di una parte delle postazioni informatiche dislocate nei vari uffici sul territorio, arrivando in un anno a sostituire 1800 vecchi computer a noleggio con altrettante postazioni di nuova fornitura, dotate di sistema operativo Linux, collegate in rete e attrezzate per le diverse necessità dei vari dipartimenti.

Tra le altre iniziative recenti degne di nota vi sono: il patrocinio da parte del Comune di Napoli degli ultimi due *Linux Day* e l'istituzione a marzo del 2011 di un Osservatorio Campano sull'Open Source. Quest'ultimo, indipendentemente dai risultati concreti che esso porterà e che saranno verificabili solo nel futuro, porta in sé un'idea interessante: un nucleo di valutazione istituito da una pubblica amministrazione composto da servizi tecnici, consiglieri comunali, ricercatori esperti di comunicazione e associazioni che promuovono il software libero, che ha il compito di giudicare l'efficacia della migrazione verso il software libero e *open source* e promuovere lo stesso processo per quel che riguarda i cosiddetti "applicativi verticali" utilizzati negli uffici dell'Ente (*database*, software di gestione dei *web server*, eccetera); un gruppo di lavoro che valuta anche le eventuali richieste d'acquisto dei software commerciali da parte dei servizi tecnici e amministrativi del Comune.

Con questa premessa, nelle pagine che seguono si cerca di mostrare come sia possibile sposare le

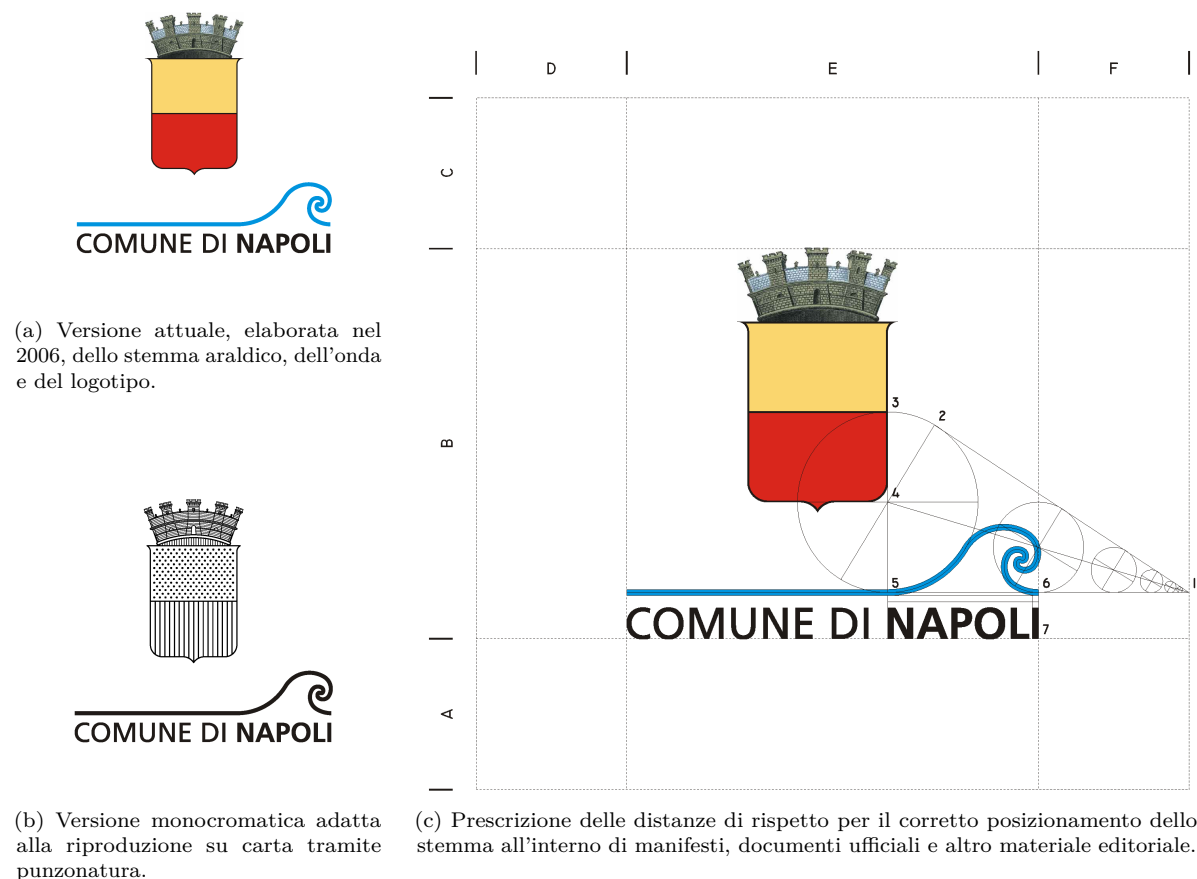


FIGURA 1: Il marchio del Comune di Napoli.

attuali possibilità offerte dal software libero e aperto con le esigenze della comunicazione istituzionale all'interno di una pubblica amministrazione. In particolare, in alternativa all'uso di programmi *open source* dedicati al *desktop publishing*, si racconteranno i motivi che hanno portato a scegliere una soluzione basata sul motore di composizione X_YL^AT_EX sperimentando nuovi modi di operare nel campo dell'*office automation*. L'esperimento è tanto più interessante se si pensa che la soluzione applicativa che viene presentata in questo articolo è potenzialmente utilizzabile da migliaia di dipendenti comunali (circa 5000) che vi hanno accesso.

Nella prossima sezione si parlerà del progetto di identità visiva del Comune di Napoli e della sua applicazione negli strumenti di comunicazione istituzionale. In particolare, l'attenzione sarà focalizzata sul formato delle lettere di comunicazione ufficiali specificato dal manuale della *Corporate Identity* dell'Ente. Nella sezione successiva si presenterà la *web application* FACILE — sviluppata utilizzando software libero o aperto — realizzata nell'ambito del progetto ADMINISTRA (“gestione elettronica degli Atti e dei Documenti amministrativi”), per la creazione guidata di lettere di comunicazione istituzionale ad uso interno (“FAi una Lettera secondo la *Corporate Identity*”). Nella parte finale dell'articolo saranno presentati alcuni aspetti tecni-

ci (e T_EXnici) legati all'integrazione dell'interfaccia utente di FACILE con il compilatore X_YL^AT_EX e con un *template* di sorgente basato sulla classe *scr1tr2*.

2 L'identità visiva del Comune di Napoli (*Corporate Identity*)

Nel Novembre 2005, l'Amministrazione comunale di Napoli indice un concorso di idee per la realizzazione della linea grafica dell'Ente. Tra le linee guida del concorso viene stabilito di mantenere inalterato lo stemma della città, innovandolo attraverso un'elaborazione del simbolo stesso che ne migliorasse la leggibilità e la riproducibilità conservando immutati gli elementi distintivi.

L'antico stemma e la nuova onda sono i due inequivocabili segni del nuovo logo del Comune di Napoli mostrato nella figura 1. Il marchio sintetizza le idee di tradizione ed innovazione, di antico e moderno, tutti elementi caratterizzanti la città di Napoli, e vuole comunicare all'osservatore la propensione a guardare al futuro senza distogliere lo sguardo dal passato, fatto di storia, cultura e tradizioni.

Il nuovo logo del Comune di Napoli ha rappresentato il punto di partenza di un progetto volto a dare alla città un vero e proprio *strumento di identificazione*, e, nel contempo, a definire la *Corporate Identity* dell'Amministrazione comunale. Nelle

TABELLA 1: Tipi di carte da lettera istituzionale, definita nella sezione “Modulistica Istituzionale” del manuale della *corporate identity* del Comune di Napoli.

Nome	Mittente	Formato	Testi (composizione intestazioni, piedini, eccetera)	Colori (marchio e testi)	Materiale Carta
Tipo A1	Il Sindaco e i suoi collaboratori (il marchio prevede la variante <i>senza onda</i> con il logotipo “Il Sindaco di NAPOLI”)	UNI A4	Adobe Garamond corsivo e nero-corsivo, Frutiger nero	marchio in colori metallici; testi in colore nero.	tipo Fedrigoni Splendorgel Extra White, di peso non inferiore a 80 g/m ²
Tipo A2	Il Sindaco e i suoi collaboratori (il marchio prevede la variante <i>senza onda</i> con il logotipo “Il Sindaco di NAPOLI”)	UNI A4	Adobe Garamond corsivo e nero-corsivo, Frutiger nero	marchio in colori metallici; testi in colore nero.	tipo Fedrigoni Corolla Classic Ivory, di peso non inferiore a 80 g/m ²
Tipo A3	Il Sindaco e i suoi collaboratori (il marchio prevede la variante <i>senza onda</i> con il logotipo “Il Sindaco di NAPOLI” ed è punzonato a secco)	UNI A4	Adobe Garamond corsivo e nero-corsivo, Frutiger nero	marchio in colori metallici; testi in colore nero.	tipo Fedrigoni Corolla Classic Premium White, di peso non inferiore a 80 g/m ²
Tipo B	Vice Sindaco, Assessori, Direttore Generale, Segretario Generale, Capo di Gabinetto, Vice Segretario Generale, Presidente e Vice Presidenti del Consiglio Comunale, Consiglieri Comunali, Presidenti di Commissione, Gruppi Consiliari	UNI A4	Adobe Garamond corsivo e nero-corsivo, Frutiger nero	marchio in ciano, magenta, giallo e nero; testi in colore nero.	tipo Fedrigoni Splendorgel Extra White, di peso non inferiore a 80 g/m ²
Tipo C	Direzioni Centrali, Dipartimenti Autonomi, Servizi Autonomi	UNI A4	Adobe Garamond corsivo e nero-corsivo, Frutiger nero	marchio in ciano, magenta, giallo e nero; testi in colore nero.	tipo Fedrigoni Splendorgel Extra White, di peso non inferiore a 80 g/m ²
Tipo D1	Presidente della Municipalità	UNI A4	Adobe Garamond corsivo e nero-corsivo, Frutiger nero	marchio in ciano, magenta, giallo e nero; testi in colore nero.	tipo Fedrigoni Splendorgel Extra White, di peso non inferiore a 80 g/m ²
Tipo D2	Altri organi di governo della Municipalità.	UNI A4	Adobe Garamond corsivo e nero-corsivo, Frutiger nero	marchio in ciano, magenta, giallo e nero; testi in colore nero.	tipo Fedrigoni Splendorgel Extra White, di peso non inferiore a 80 g/m ²

scienze della comunicazione la *corporate identity* (“identità societaria” o anche “identità visiva”) è intesa come una strategia condivisa che unifichi per l’intera azienda il modello comunicativo, sia al suo interno che verso l’esterno.

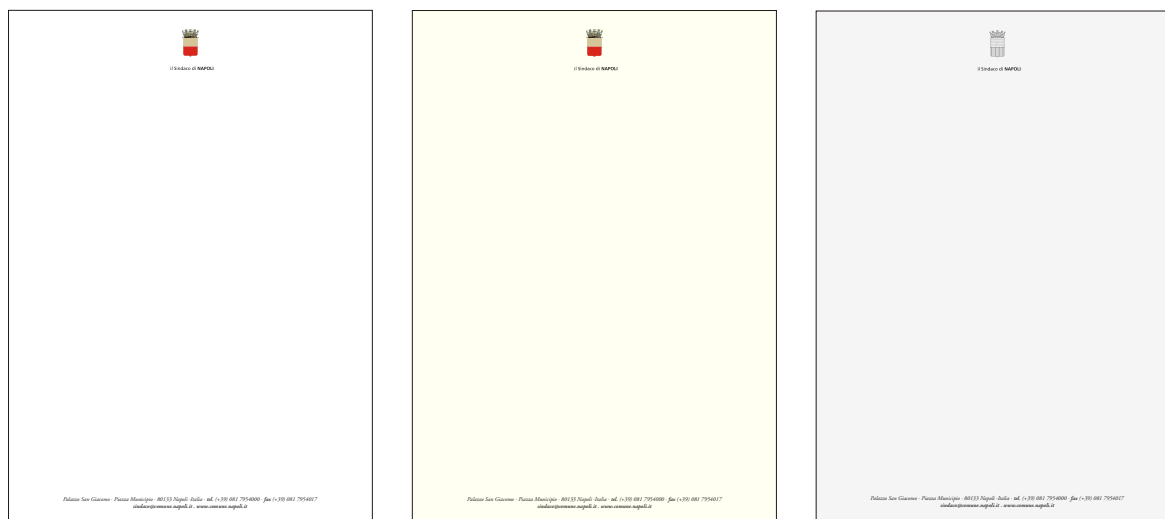
L’idea alla base dell’attuale identità visiva del Comune di Napoli è quella di difendere, aggiornandola, l’identità di una città, facendone strumento di proiezione in ambito euromediterraneo. Il progetto grafico si è mosso su due direttrici: (i) il recupero dello stemma della città, con un intervento mirato ad assicurarne la massima leggibilità, l’adozione del logotipo “COMUNE DI NAPOLI” e l’utilizzo dell’onda come segno; (ii) la definizione e l’unificazione dei materiali di comunicazione dell’azienda comunale. È proprio quest’ultimo punto il motivo ispiratore del presente articolo.

2.1 Il manuale della *Corporate Identity* del Comune di Napoli

La creazione della *corporate identity* dell’Ente risale al 2006 ed è documentata in un apposito manuale (ANONIMO, 2006) che presenta gli elementi

di identità visiva del Comune di Napoli. Il suo scopo è quello di fornire una guida agli addetti ai lavori — dipendenti e consulenti — affinché si superi la frammentazione nei materiali e nel segno grafico riscontrata nel lavoro di uffici e assessorati e si operi secondo un’unica strategia comunicativa, approvata ed adottata dall’intero Ente.

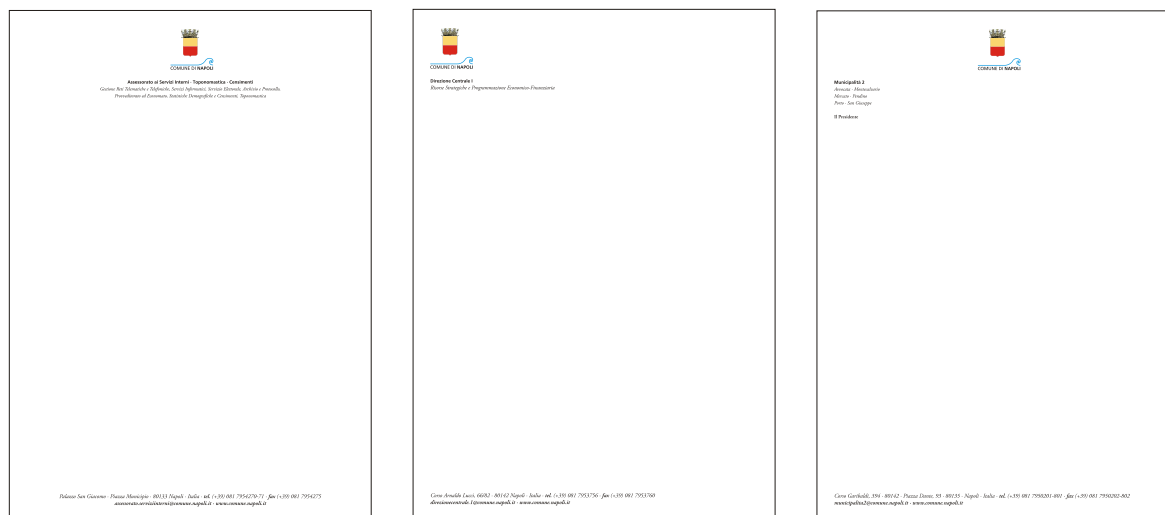
La linea grafica del Comune di Napoli, e la sua applicazione nei diversi strumenti di comunicazione interna ed esterna all’Ente, *deve* essere rigorosamente rispettata, così come definita nel manuale, senza possibilità di adattamenti e interpretazioni personali. Chiunque debba operare oggi — dipendenti comunali o consulenti che si accingono a preparare lettere, manifesti o altre forme ufficiali di comunicazione visiva — su un ambito, anche minimo, del sistema stesso, trova nelle tavole del manuale gli strumenti adatti. Il manuale è uno strumento nel quale cercare gli elementi che servono, trovandoli direttamente o cercando nella stessa famiglia la soluzione allo specifico problema.



(a) Lettera del Sindaco (Tipo A1).

(b) Lettera del Sindaco (Tipo A2).

(c) Lettera del Sindaco (Tipo A3).



(d) Lettera dell'Assessorato ai Servizi Interni, Toponomastica e Censimenti (Tipo B).

(e) Lettera della Direzione Centrale I, Risorse Strategiche e Programmazione Economico-Finanziaria (Tipo C).

(f) Lettera del Presidente della Municipalità 2 (Tipo D).

FIGURA 2: Esempi di carta da lettera per comunicazioni ufficiali del Comune di Napoli (si veda anche la tabella 1). Testatina e piedini composti diversamente a seconda del tipo di comunicazione.

2.2 Lo stemma del Comune di Napoli

Il marchio rappresentativo del Comune di Napoli è costituito da tre elementi, combinati tra loro secondo un preciso schema grafico. Questi elementi sono: (i) l'antico stemma; (ii) il nuovo simbolo, l'onda; (iii) il logotipo "COMUNE DI NAPOLI".

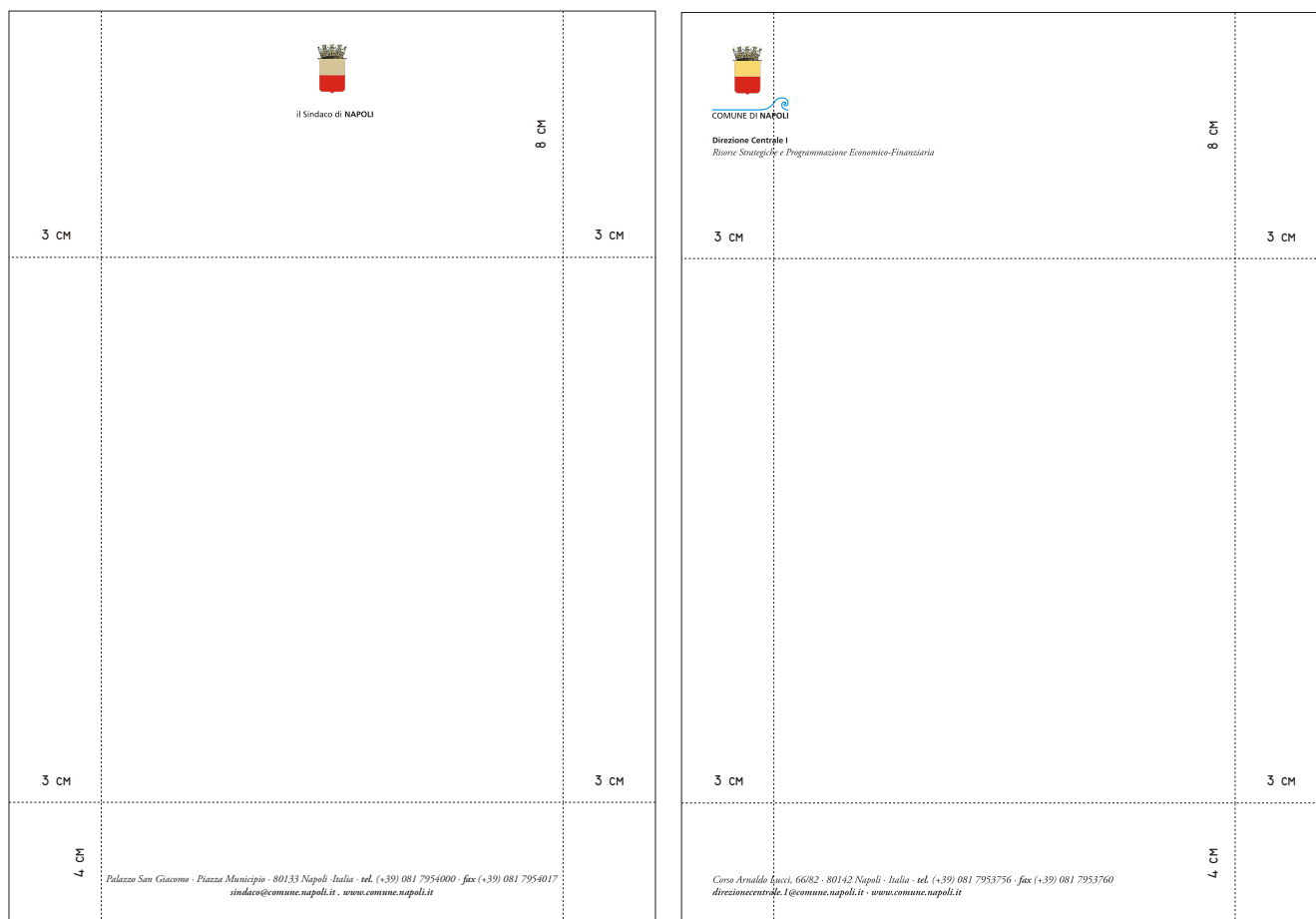
L'elemento più importante e caratterizzante è lo stemma, risalente al XIV secolo¹. Una rappresentazione dello stemma è mostrata nella figura 1. Nella figura 1a si riporta la versione ufficiale, a colori. Nella figura 1b viene mostrata una variante monocromatica destinata a particolari utilizzi. La posizione e la distanza tra gli elementi che compongono il marchio è fissa e invariabile. Nella figura 1c

1. Si vedano: http://it.wikipedia.org/wiki/Stemma_di_Napoli ed <http://www.comune.napoli.it/flex/cm/pages/ServeBLOB.php/L/IT/IDPagina/3504>.

sono mostrate le distanze di rispetto prescritte nel manuale dell'identità visiva per il corretto posizionamento dello stemma all'interno di manifesti, documenti ufficiali e altro materiale editoriale.

Il marchio, in tutte le versioni d'uso, non può essere riprodotto con una misura di base inferiore a 24,7 mm (la misura fa riferimento alla larghezza del logotipo). Al di sotto di tale misura gli elementi che compongono il marchio perdono infatti di leggibilità.

Per quanto riguarda il cosiddetto *lettering* istituzionale, come carattere del marchio è stato scelto il font Frutiger (un font senza grazie), negli stili tondo e nero. Nel logotipo la parte "COMUNE DI" è in tondo e la parte "NAPOLI" è in nero. Come estensione consentita del carattere istituzionale è possibile utilizzare unicamente il font Frutiger



(a) Lettera del Sindaco (Tipo A1).

(b) Lettera da parte di Direzioni Centrali, Dipartimenti Autonomi, Servizi Autonomi (Tipo C).

FIGURA 3: Modelli di carta da lettera per comunicazioni ufficiali del Comune di Napoli. Definizione della gabbia del testo (si veda anche la tabella 1).

nello stile corsivo. Non è consentito l'utilizzo di altri caratteri. Per garantire una buona leggibilità è stato individuato il corpo 7,5 come dimensione minima, in riferimento al formato UNI A4, cioè il formato di carta maggiormente usato.

2.3 Modelli di carta da lettera

Il manuale dell'identità visiva dell'Ente definisce in maniera sistematica i formati ammissibili delle comunicazioni ufficiali, destinate sia alla stampa che all'archiviazione elettronica. Il recapito delle comunicazioni, a seconda della tipologia e dei destinatari, in genere viene effettuato per via postale oppure per via elettronica.

Le comunicazioni istituzionali possono avere uno o più destinatari ('principali' e 'in copia') e si distinguono in due categorie: le comunicazioni interne e le comunicazioni esterne. Le comunicazioni interne sono destinate ad uso interno dell'Ente e riguardano tipicamente comunicazioni tra dipartimenti, uffici, eccetera. Le comunicazioni esterne sono dirette da un soggetto istituzionale appartenente all'Amministrazione comunale a un destinatario esterno all'Ente, istituzionale anch'esso o privato.

Il quadro dei tipi di lettera istituzionale definiti nel manuale della Corporate Identity del Comune di Napoli è riportato nella tabella 1. Alcuni esempi di lettera — un foglio bianco con testatina e piedini composti diversamente a seconda del tipo di comunicazione — sono mostrati nella figura 2. Nella figura 3 sono riportati due esempi tratti dal manuale dell'identità visiva in cui sono specificate nel dettaglio le dimensioni dei margini che individuano la gabbia del testo.

Le categorie di comunicazione vengono definite in base al mittente (Tipo A1, A2 A3, B, C, D1, D3). L'applicazione FACILE (che sarà descritta più avanti) è stata progettata in base all'esigenza iniziale di uniformare i formati delle comunicazioni *interne*. Essa introduce una procedura guidata attraverso un'interfaccia utente (una *web form*) che consente la creazione di lettere istituzionali di Tipo C con destinatari interni all'ente. Questa limitazione è dovuta al carattere innovativo e sperimentale dell'applicazione. Si è voluto restringere il campo dei possibili tipi di documento da produrre in maniera automatica, per poi passare ad una fase successiva di 'sperimentazione sul campo' ed infine valutare



FIGURA 4: La pagina iniziale della intranet del Comune di Napoli. L'applicazione FACILE è integrata nel menu a disposizione dei dipendenti.

l'efficacia dello strumento di lavoro.

La modulistica di Tipo C è prevista per le comunicazioni interna tra Direzioni Centrali, i Dipartimenti Autonomi e i Servizi Autonomi del Comune. L'impostazione grafica prevede il posizionamento a bandiera a sinistra del marchio ed una distribuzione a bandiera a sinistra dei testi all'interno della testatina e del piedino. La comunicazione deve avere un formato di pagina di tipo UNI A4. Il font stabilito per il testo nella testatina e nel piedino è Adobe Garamond (corsivo e nero-corsivo) e Frutiger (tondo e nero). Il font del testo della comunicazione è Adobe Garamond. Per garantire una buona leggibilità è stato individuato il corpo 10. Non è consentito l'utilizzo di altre famiglie di caratteri. La riproduzione del marchio corrisponde a quella riportata nella figura 1a. Per le versioni a stampa deve essere usata carta di tipo Fedrigoni Splendorgel Extra White, di peso non inferiore a 80 g/m², idonea ad essere stampata a più colori con tecnologia *offset* oppure dalle comuni stampanti laser.

Per quel che concerne l'impostazione della gabbia di testo nella pagina, per una lettera di Tipo C sono da rispettare i seguenti margini: un margine superiore di 8 cm, un margine inferiore di 4 cm, un margine destro di 3 cm, un margine sinistro di 3 cm. In generale i fogli delle lettere più lunghe di una pagina portano il testo solo sulle pagine dispari (*recto*).

In casi eccezionali, il sistema di comunicazione del Comune di Napoli, accanto ai caratteri Frutiger e Adobe Garamond, prevede una possibile estensione del *lettering* istituzionale. I font ulti-

mali possono essere sostituiti con i seguenti font compatibili: Verdana per quel che riguarda l'estensione del *lettering* del marchio, Verdana e Times New Roman per l'estensione del *lettering* della modulistica.

3 La web application FACILE

L'idea che ha ispirato il progetto di FACILE era rivolta al dipendente comunale addetto alla preparazione e all'invio di una comunicazione interna protocollata, invogliandolo all'uso di uno strumento diverso da un programma di video scrittura. Una motivazione iniziale era costituita dalla volontà di evitare che il singolo utente prendesse iniziative personali sul modo in cui comporre una lettera — usando eventualmente font non consentiti o vecchie versioni del logo dell'Ente oppure, ancora, riciclando il formato di vecchi documenti non aderenti alle specifiche del manuale della *corporate identity*.

In secondo luogo, considerato che un lavoratore comunale normalmente usa e gestisce la posta attraverso un *browser* collegato alla Intranet della pubblica amministrazione, è stato ritenuto un aspetto molto importante per la fortuna di questa sperimentazione la necessità di dare all'utente la percezione che la composizione di una comunicazione interna fosse tanto semplice quanto gestire la posta elettronica.

Pertanto la scelta progettuale è caduta su uno strumento di lavoro simile a una *webmail*, con la differenza che il risultato finale sarebbe dovuto essere un file in formato PDF da stampare (e successivamente protocollare), archiviare e diramare. Data la

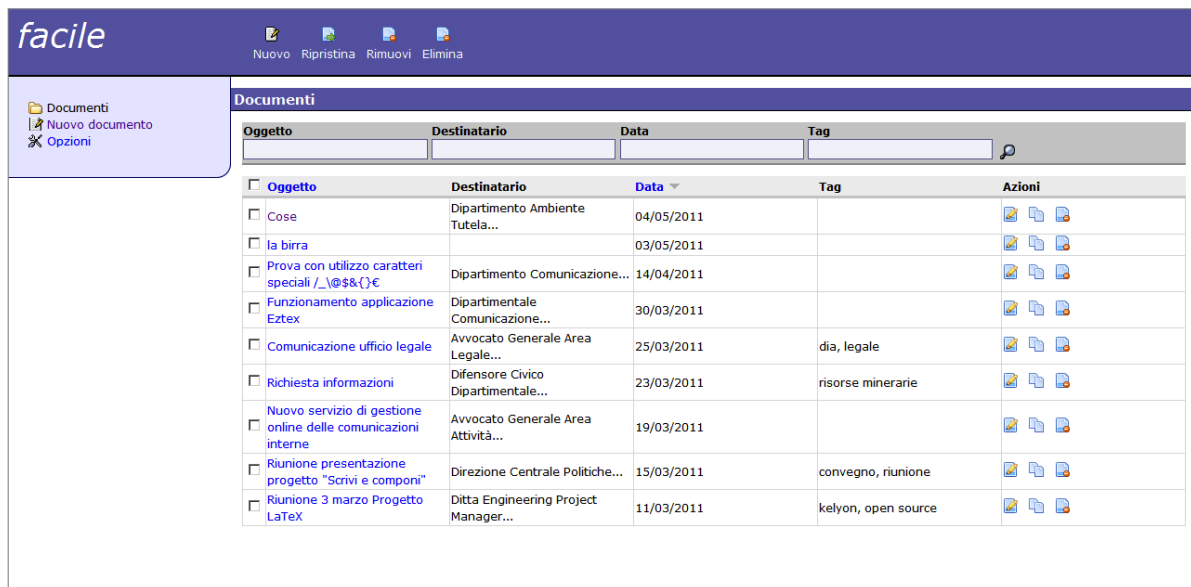


FIGURA 5: La pagina iniziale dell'applicazione FACILE.

necessità di usare i font prescritti dal manuale della *Corporate Identity* del Comune di Napoli, una soluzione basata sulle possibilità offerte da XELATEX, opportunamente “occultato” offrendo all’utente un *front-end* semplificato, è apparsa dunque la via ideale da percorrere.

Un evidente vantaggio derivante da questa impostazione è quello di poter avere uno strumento di lavoro condiviso, basato sull’esistenza di servizi informatici e archivi centralizzati a disposizione della platea di utenti, puntando a eliminare le differenze tra le prassi lavorative riscontrate nei diversi uffici.

3.1 Caratteristiche principali

La filosofia di sviluppo alla base di FACILE parte dalla considerazione che è necessario gestire migliaia di utenti che utilizzano sia sistemi operativi Windows che, in misura significativa, Linux (grazie alla migrazione voluta dal Comune di Napoli negli ultimi anni). L’esigenza che deriva da queste considerazioni ha spinto a sviluppare un sistema *server based*. Questo approccio semplifica drasticamente la gestione dei modelli da adottare e il flusso delle comunicazioni da produrre, attraverso la gestione centralizzata degli accessi, che vengono legati alla Intranet dell’Ente, e delle politiche di redazione delle comunicazioni (tipo di modello utilizzato dal singolo dipendente, flusso autorizzativo, aggiornamento dei modelli).

Si è scelto di sviluppare FACILE utilizzando il *Content Management System* (CMS) *open source* Drupal². Questo *framework* di sviluppo ha permesso di implementare in tempi ridotti il *front-end* dell’applicazione e di gestire agevolmente l’inserimento dei contenuti delle comunicazioni, dei desti-

nari (da scegliere a partire da liste preimpostate e memorizzate in un *database* comunale) e dei parametri necessari al completamento della redazione e della spedizione dei documenti.

Si può immaginare FACILE come un’applicazione costituita da tre ‘strati’ software. Lo strato più esterno è l’interfaccia grafica, che è percepita dall’utente come una pagina web. Uno strato software molto importante è quello intermedio che raccoglie tutti i contenuti inseriti attraverso l’interfaccia grafica e li mette in relazione con le funzionalità di composizione del documento finale. Questo modulo del sistema non è altro che un *parser*, sviluppato in linguaggio PHP. Il suo scopo è di elaborare i contenuti inseriti dall’utente. Il risultato dell’elaborazione viene copiato al posto di alcune stringhe chiave contenute in un *template* di sorgente predisposto per essere compilato con XELATEX. Quest’ultimo rappresenta il cuore del terzo strato software, quello più interno.

Già in fase di progettazione e successivamente nel corso dello sviluppo dell’interfaccia grafica e del *parser* è stato utilizzato un approccio teso alla generalizzazione dei casi d’uso, prevedendo estensioni future dell’applicazione. A questo scopo sono stati definiti vari parametri di configurazione che rendono l’applicazione flessibile e consentono l’aggiunta di funzionalità supplementari senza modificarne la logica. Ad esempio, in un prossimo futuro i documenti generati con FACILE potranno essere firmati digitalmente, protocollati, inviati automaticamente ai destinatari via posta elettronica e archiviati nel sistema documentale del Comune di Napoli.

Il primo rilascio di FACILE risale all’aprile 2011. L’applicazione è stata installata su uno dei server dell’Intranet del Comune di Napoli insieme con la distribuzione TEX Live 2010.

2. drupal.org.

facile

- Documenti
- Nuovo documento
- Opzioni

Crea Firma

Titolo:

Nome:

Cognome:

Dipartimento:

Servizio:

Ruolo:

Indirizzo:

Telefono:

Fax:

E-mail:

Sito web:

Salva

FIGURA 6: La pagina di configurazione della firma.

facile

- Documenti
- Nuovo documento
- Opzioni

Crea nuovo documento

Seleziona il modello da creare

Categoria: *
Comunicazioni

Sottocategoria:
Interne

Modello: *
Comunicazione interna standard

Crea

FIGURA 7: Schermata con la quale l'utente sceglie di creare un nuovo documento.

3.2 Breve panoramica sulle funzionalità

Per accedere a FACILE l'utente utilizza un *browser*, collegandosi alla pagina principale della Intranet del Comune di Napoli. Qui è disponibile una *tab* dedicata all'applicazione, come si vede dalla figura 4. Cliccando sulla *tab* l'utente, qualora autorizzato, accederà direttamente all'applicazione.

La pagina di accesso a FACILE è mostrata nella figura 5. La schermata principale di FACILE è com-

posta da un menu sul lato sinistro della finestra e da un'area di lavoro costituita dalla parte rimanente della schermata. Il menu permette all'utente di creare un nuovo documento (cliccando sulla voce "Nuovo documento") e di gestire le proprie opzioni (voce "Opzioni"). Se l'utente ha già creato documenti, sul lato destro della schermata principale ne sarà visualizzato l'elenco recante l'oggetto, il destinatario, la data e i "tag" (ovvero delle categorie per organizzare e gestire con semplicità i documenti).

Nella parte alta della schermata principale sono presenti alcune icone per la gestione rapida di azioni sui documenti (modifica, duplicazione, eliminazione, eccetera).

Dal menu principale attraverso la selezione della scelta "Opzioni" si passa a delle schermate di configurazione del profilo utente. Un esempio è la schermata mostrata nella figura 6 con la quale è possibile definire una *firma*. La firma è importante poiché i dati ad essa associati vengono utilizzati, unitamente al logo dell'Ente, nella testatina e nel piede delle lettere. Ciascun utente può definire più di una firma — caratteristica utile quando la lettera viene materialmente preparata da una persona diversa da quella che compare ufficialmente come mittente. Ciascun utente ha la possibilità di memorizzare più di una firma all'interno del proprio profilo, scegliendo di volta in volta quella

appropriata alla lettera da mandare in stampa.

Cliccando sulla voce “Nuovo documento” del menu principale si accede ad una schermata come quella della figura 7. Questo passaggio consentirà in futuro di scegliere tra diversi tipi di documento, a seconda se la comunicazione è ad uso interno o esterno. Attualmente con FACILE è possibile comporre unicamente lettere di comunicazione ufficiali ad uso interno.

Cliccando sul pulsante “Crea” nella schermata della figura 7 si accede ad una schermata come quella visualizzata nella figura 8. All’utente viene presentato un *form* con un numero di campi da riempire. La struttura di questa finestra — una delle più importanti — rispecchia quella di una lettera. Sono presenti tutti gli elementi di cui ci si preoccupa quando si prepara una comunicazione ufficiale da diramare eventualmente a più destinatari.

Esaminando i diversi elementi della finestra ci si rende conto che si tratta di una sorta di *editor* con funzionalità particolari. Ad esempio, il primo campo è denominato “Intestazione A:”. Esso corrisponde all’etichetta che si vuole utilizzare nella lettera per indicare il destinatario (solitamente l’etichetta consiste in “A:”, ma potrebbe essere anche “Ai dirigenti:”). il campo successivo denominato “A:”, invece, identifica il destinatario vero e proprio (ad esempio: “Tizio Caio, Via Tal dei Tali 123, 80100 Napoli”). Per inserire un destinatario si può procedere sia manualmente, riempiendo la casella di testo corrispondente, oppure, cliccando sul segno “+” a destra del campo “A:”. In questo caso si ha accesso ad una rubrica residente sul server dell’applicazione, un *database* contenente le diciture ufficiali di tutti i possibili destinatari con i loro indirizzi. La rubrica può essere scorsa e l’utente può selezionare il destinatario cercato. I destinatari di questa sezione possono essere molteplici.

Un discorso analogo al precedente vale per il campo “CC:” riferito ai destinatari a cui inviare una copia in conoscenza del documento. Il campo “Intestazione CC:” corrisponde all’etichetta che si vuole utilizzare sul documento per indicare i destinatari secondari (solitamente l’etichetta consiste in “E, p. c.:”). Il campo “CC:”, invece, identifica i destinatari secondari. Per inserire un destinatario a cui inviare una copia in conoscenza del documento si può procedere sia manualmente, riempiendo la casella di testo corrispondente, oppure, cliccando sul segno “+” a destra del campo “CC:”, scorrendo la rubrica dei contatti e selezionando il destinatario.

Dalle descrizioni precedenti è intuitivo comprendere il significato dei campi successivi: “Luogo”, “Data”, “Oggetto” e così via.

La sezione centrale del *webform* è il campo denominato “Corpo”. Il corpo del documento è un vero e proprio *editor* che consente di inserire il testo della lettera e di formattarlo con gli appositi pulsanti. Il componente software che espone questa

funzionalità³ traduce il materiale immesso dall’utente in linguaggio HTML. Il *parser* di FACILE — lo strato software intermedio — raccoglierà questo codice traducendolo in codice L^AT_EX.

È quest’ultima una funzionalità sviluppata appositamente con l’applicazione. Attualmente tutti i principali costrutti di *markup* del linguaggio HTML sono tradotti nel *markup* del formato L^AT_EX (con possibilità di inserire immagini e di comporre correttamente anche delle tabelle non troppo complesse).

Una volta inserito il testo del documento, si può inserire una formula di chiusura, dopodiché è necessario selezionare la propria firma dall’apposito menu a tendina. I dati della firma vengono utilizzati nella intestazione della lettera, nell’area della firma e nel piedino del documento stesso.

Qualora si volessero aggiungere degli allegati al documento, è possibile cliccare sulla voce “File allegati”, quindi selezionare uno o più file. Questa sezione permette di far comparire una lista eventuale di allegati in calce alla firma. L’utente ha la possibilità di inserire il testo che descrive i singoli allegati (che andrà effettivamente composto) e al tempo stesso di caricare i file corrispondenti. Questi ultimi saranno archiviati insieme a tutti i dati associati al documento.

Cliccando sul pulsante “Salva” sarà possibile salvare il lavoro svolto fino a quel momento e continuare l’inserimento dei dati del documento corrente senza creare il relativo file PDF. Per visualizzare l’anteprima del documento sarà possibile cliccare su “Anteprima”. Altrimenti, per completare la procedura di creazione del documento, basterà cliccare sul pulsante “Salva e completa”. In questo modo verrà creato il file PDF del documento corrente, scaricabile sul computer dell’utente e stampabile.

Quando l’utente termina il suo lavoro con la schermata della figura 8 il contenuto dei campi viene raccolto ed elaborato dal *parser*. Quando viene premuto il pulsante “Salva e completa” (oppure “Anteprima”) il *parser* traduce l’input dell’utente e al tempo stesso accede ad un file `template.tex`. Quest’ultimo è memorizzato sul server dell’applicazione ed è unico per tutti gli utenti. Il *template* viene usato come base per la generazione di un file sorgente `<nome-lettera>.tex`. Il nome unico del file sorgente viene costruito in base all’oggetto della lettera e alla data di creazione. Questo file non è accessibile agli utenti normali ma solo agli utenti con privilegio di amministratore. Ai primi sarà visibile solo il risultato della compilazione del sorgente con il programma `xelatex`, cioè il file `<nome-lettera>.pdf`. Agli utenti amministratori sarà concesso sia di caricare eventuali aggiornamenti del file `template.tex` che accedere ai sorgenti delle singole lettere.

3. Si veda: <http://ckeditor.com>.

Comunicazione interna standard -- Formattazione corretta dicitura servizio

Intestazione A: *

A:

A: *

DCFP1025

Amministrazione delle Risorse Umane – Area Giuridica

LUCIA DI MICCO

+

-

+

-

Aggiungi un'altra voce

Intestazione CC:

E, p.c.:

CC:

DCFP8005

U.O.A. Elaborazione degli Stipendi

GIOACCHINO CATUOGNO

+

-

+

-

Aggiungi un'altra voce

Luogo:

Data:

Oggetto: *

Comunicazione interna standard -- Formattazione corretta dicitura servizio

Corpo:

Codice Sorgente

Normale

B

I

vi comunico che il 31/08 ultimo scorso sono state avviate le procedure per l'unificazione dei formati elettronici ...

body p

Passa all'editor di solo testo

Formula chiusura:

Cordialità

Firma: *

Ing. SCARDI Antonio Dipartimento Programmazione Economica e del Territorio Servizio Economato e Ragione

Crea source LaTeX

File allegati

Tag documento

Tag assegnati:

+

Clicca sul pulsante a fianco o premi invio per aggiungere un tag alla lista. I tag sono dei termini che descrivono il contenuto del documento (es. amministrazione, pratica, comunicazione interruzione servizio).

Salva

Salva e completa

Anteprima

Rimuovi

Elimina

FIGURA 8: Creazione di un documento.



Dipartimento Comunicazione Istituzionale, Tecnologie e Società dell'informazione
Servizio SIAD – Sistema Informativo, Amministrativo e Documentale

Prot.



A: Direzione Centrale I – Risorse strategiche e programmazione economico-finanziaria
Servizio SIF – Sistema Informativo Finanziario
 dott. Umberto MANDARA

Servizio RTI e Microinformatica
SEDE
 dott.ssa Rosanna PERSICO

Data
 31/08/2011



E, p.c.: Direzione Generale
Servizio Controllo qualità
 dott. Massimo PACIFICO
Servizio Portale Web e nuovi media
SEDE
 ing. Giuseppe CONTINO

Oggetto: Attività di sperimentazione della web application FACILE

Carissimi,

In merito alle attività in oggetto, che avranno inizio il giorno 23 settembre 2011, vi prego di dare massima diffusione possibile degli allegati alla presente.

Cordialità,

ing. Paolo Eugenio CRESCI
Responsabile Servizio Informatico

Allegati:

Allegato A. Regolamento e manuale d'uso della Intranet del Comune di Napoli. Aggiornamenti 2011.

Allegato B. Manuale utente dell'applicazione FACILE.

*Piazza Giovanni XXIII n. 6, 80126 Napoli, Italia · tel. (+39) 081 7958800 · fax (+39) 081 7958803
 sistema.informativo.amministrativo@comune.napoli.it · www.comune.napoli.it*

FIGURA 9: Esempio di lettera per comunicazione interna prodotto con l'applicazione FACILE.

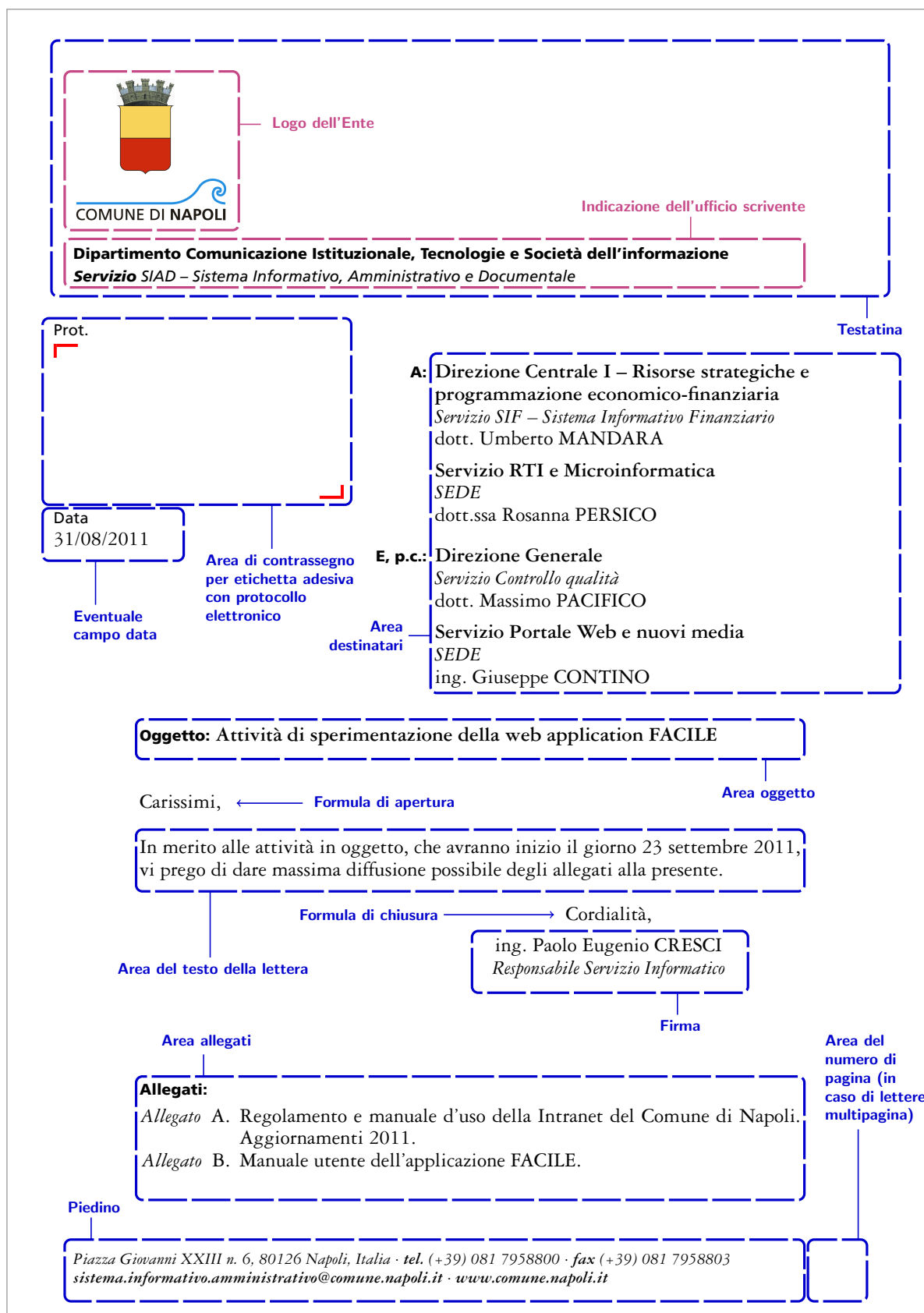


FIGURA 10: La medesima lettera mostrata nella figura 9. Sono evidenziati gli elementi che compongono la struttura tipografica del documento.



COMUNE DI NAPOLI

Dipartimento Comunicazione Istituzionale, Tecnologie e Società dell'informazione
Servizio SIAD - Sistema Informativo, Amministrativo e Documentale

Prot. ┐

A: Assessorato alle politiche sociali
Resp. Quarta Vicaria, dott.ssa Rosa Russo
 Piazza Vicario, 31bis
 Comando VV. UU.
Com. Luigi Fariello
 Ufficio Stampa
 Comando VV. FF.
Com. ing. Tiberio Sauro
 Sede

E, p.c.: Assessorato alle politiche sociali
Resp. Quarta Vicaria, dott.ssa Rosa Russo
 Piazza Vicario, 31bis
 Comando VV. UU., Sezione S. Ferdinando - Chiaia
 - Posillipo - Bagnoli - Fuorigrotta
Com. Luigi Fariello
 Presidenza del Polo delle scienze
prof. ing. Paolo Saladino
 Via Claudio, 31 - 80125 Napoli
 Politecnico Gioacchino Murat
prof. Fulvio Testi
 Rettorato

Oggetto: Chiarimenti sulle nuove norme in merito allo smaltimento di acque reflue

Cari tutti,

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetur id, vulputate a, magna. Donec vehicula augue eu neque. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Mauris ut leo. Cras viverra metus rhoncus sem. Nulla et lectus vestibulum urna fringilla ultrices. Phasellus eu tellus sit amet tortor gravida

Piazza Giovanni XXIII n. 6, 80126 Napoli, Italia - tel. (+39) 081 7958800 - fax (+39) 081 7958803
 sistema.informativo.amministrativo@comune.napoli.it - www.comune.napoli.it

Pag. 1 di 3

(a) pagina 1/3



COMUNE DI NAPOLI

placemat. Integer sapien est, iaculis in, pretium quis, viverra ac, nunc. Praesent eget sem vel leo ultrices bibendum. Aenean faucibus. Morbi dolor nulla, malesuada eu, pulvinar at, mollis ac, nulla. Curabitur auctor semper nulla. Donec varius orci eget risus. Duis nibh mi, congue eu, accumsan eleifend, sagittis quis, diam. Duis eget orci sit amet orci dignissim rutrum.

Nam dui ligula, fringilla a, euismod sodales, sollicitudin vel, wisi. Morbi auctor lorem non justo. Nam lacus libero, pretium at, lobortis vitae, ultricies et, tellus. Donec aliquet, tortor sed accumsan bibendum, erat ligula aliquet magna, vitae ornare odio metus a mi. Morbi ac orci et nisl hendrerit mollis. Suspendisse ut massa. Cras nec ante. Pellentesque a nulla. Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Aliquam tincidunt urna. Nulla ullamcorper vestibulum turpis. Pellentesque cursus luctus mauris.

Nulla malesuada porttitor diam. Donec felis erat, congue non, volutpat at, tincidunt tristique, libero. Vivamus viverra fermentum felis. Donec nonummy pellentesque ante. Phasellus adipiscing semper elit. Proin fermentum massa ac quam. Sed diam turpis, molestie vitae, placerat a, molestie nec, leo. Maecenas lacinia. Nam ipsum ligula, eleifend at, accumsan nec, suscipit a, ipsum. Morbi blandit ligula feugiat magna. Nunc eleifend consequat lorem. Sed lacinia nulla vitae enim. Pellentesque tincidunt purus vel magna. Integer non enim. Praesent euismod nunc eu purus. Donec bibendum quam in tellus. Nullam cursus pulvinar lectus. Donec et mi. Nam vulputate metus eu enim. Vestibulum pellentesque felis eu massa.

Quisque ullamcorper placerat ipsum. Cras nibh. Morbi vel justo vitae lacus tincidunt ultrices. Lorem ipsum dolor sit amet, consectetur adipiscing elit. In hac habitasse platea dictumst. Integer tempus convallis augue. Eriam facilisis. Nunc elementum fermentum wisi. Aenean placerat. Ut imperdiet, enim sed gravida sollicitudin, felis odio placerat quam, ac pulvinar elit purus eget enim. Nunc vitae tortor. Proin tempus nibh sit amet nisl. Vivamus quis tortor vitae risus porta vehicula.

Fusce mauris. Vestibulum luctus nibh at lectus. Sed bibendum, nulla a faucibus semper, leo velit ultricies tellus, ac venenatis arcu wisi vel nisl. Vestibulum diam. Aliquam pellentesque, augue quis sagittis posuere, turpis lacus congue quam, in hendrerit risus eros eget felis. Maecenas eget erat in sapien mattis porttitor. Vestibulum porttitor. Nulla facilisi. Sed a turpis eu lacus commodo facilisis. Morbi fringilla, wisi in dignissim interdum, justo lectus sagittis dui, et vehicula libero dui cursus dui. Mauris tempor ligula sed lacus. Duis cursus enim ut augue. Cras ac magna. Cras nulla. Nulla egestas. Curabitur a leo. Quisque egestas wisi eget nunc.

Pag. 2 di 3

(b) pagina 2/3



COMUNE DI NAPOLI

Nam feugiat lacus vel est. Curabitur consectetur.

Cordialità,
 ing. Paolo Eugenio CRESCI
 Responsabile Servizio Informatico

Allegati:
 Allegato A. Normativa comunale in merito allo smaltimento di acque reflue.
 Allegato B. Circolare ministeriale sulle visite ispettive nelle sedi di enti pubblici, amministrazioni locali, caserme.
 Allegato C. Descrizione dell'ultimo allegato.

Pag. 3 di 3

(c) pagina 1/3

FIGURA 11: Esempio di lettera multipagina prodotta con l'applicazione FACILE.

Nella figura 9 è mostrato un esempio di lettera monopagina prodotto con FACILE. La figura 10 mostra la stessa lettera e mette in evidenza i diversi elementi tipografici. Se essa viene confrontata con la schermata di FACILE della figura 8 si comprende quanto sia importante che il *parser* dell'applicazione sia efficiente e che il *template* di sorgente L^AT_EX sia in grado di restituire risultati ben composti per un ventaglio potenzialmente molto ampio di casi d'uso.

La figura 11 è un esempio di lettera lunga più di una pagina. Nel piedino, a destra, è riportata la numerazione riferita al numero totale di pagine ("1 di 3", "2 di 3", "3 di 3"). La gestione di questa specifica, oltre al caricamento di un apposito pacchetto, richiede di gestire delle compilazioni multiple con X_YL^AT_EX.

Il numero di compilazioni è un tipo di configurazione impostabile dagli utenti di FACILE con privilegio di amministratore dell'applicazione. Gli utenti amministratori hanno accesso ad un menu "Opzioni" più ricco di voci. Inoltre, un utente amministratore può anche scegliere di far restituire all'applicazione il sorgente L^AT_EX, un file .tex anziché il file in formato PDF. Ciò è utile quando l'amministratore vuole effettuare un *debug* del processo di composizione.

3.3 Il file *template* di una comunicazione interna

Il *template* di sorgente L^AT_EX che viene elaborato dal *parser* di FACILE è una parte fondamentale dell'applicazione. Nella sua attuale versione, esso consta di circa 1500 linee di testo, che comprende il codice vero e proprio e le linee di commento. Il commento del codice è stato inserito con lo scopo di fornire una guida rapida agli utenti esperti che vogliono effettuare operazioni di *debug*. I commenti facilitano la lettura del codice in generale ma, soprattutto, documentano le svariate macro presenti nel *template*, alcune delle quali scritte utilizzando comandi primitivi del linguaggio T_EX.

Il sorgente è basato sulla classe `scrlltr2` appartenente al *bundle* di pacchetti di estensione KOMA-Script (KOHM e MORAWSKI, 2011). I pacchetti di KOMA-Script sono stati sviluppati dagli autori tedeschi Markus Kohm e Jens-Uwe Morawski e sono stati scelti in base al loro alto livello di configurabilità, alla robustezza del codice e in base alla ampia comunità di utenti sparsi in tutto il mondo. Questo è un aspetto importante perché assicura un'ampia base di conoscenza con buone possibilità di trovare soluzioni a problemi specifici⁴.

3.3.1 Una implementazione basata sulla classe *scrlltr2*

Tra le classi definite da KOMA-Script vi è la `scrlltr2`, dedicata alla composizione di lettere sia private

che commerciali.

Rilasciata nel 2002, `scrlltr2` è una classe che reimplementa *ex novo* le funzionalità della classe standard `letter` di L^AT_EX. Anche se parte del codice di `scrlltr2` è identico a quello delle principali classi di KOMA-Script, le lettere sono molto diverse dagli articoli, dai saggi o dai libri. Ciò giustifica la presenza nel *bundle* KOMA-Script di una classe espressamente dedicata alle lettere. Inoltre, essendo stata progettata a partire da zero, `scrlltr2` fornisce un'interfaccia d'uso alquanto particolare rispetto alle normali classi scritte per L^AT_EX; interfaccia che conferisce alla classe una grande flessibilità.

Una caratteristica particolare della classe `scrlltr2` è data dalla possibilità di cambiare molte opzioni, anche dopo il caricamento con `\documentclass`. Il comando `\KOMAOPTIONS` serve a tal fine, accettando una lista di opzioni assegnate secondo lo schema $\langle \textit{chiave} \rangle = \langle \textit{valore} \rangle$ (grazie al pacchetto `keyval`).

Si rimanda il lettore al manuale d'uso della classe per approfondimenti sulle sue opzioni di personalizzazione.

Va osservato che il sorgente che deve scaturire dal *template* deve poter essere usato per comporre lettere di una o di più pagine. Inoltre non sono rari i casi in cui le comunicazioni vanno diramate a più di un destinatario principale ed in copia per conoscenza a numerosi destinatari secondari. Può capitare in alcuni casi che la lista completa dei destinatari preveda anche più di un salto di pagina.

Grazie a delle macro appositamente scritte e a delle particolari impostazioni della classe `scrlltr2`, il codice del *template* implementa un modello generale di lista dei destinatari con eventuali salti pagina. Per una lista di destinatari di dimensioni maggiori dello spazio disponibile sulla prima pagina il contenuto della lista viene spezzato in più parti; la prima parte viene composta nella prima pagina, la seconda parte sulla seconda pagina ed eventualmente le parti rimanenti sulle pagine successive.

Dopo la lista dei destinatari verranno collocati l'oggetto della comunicazione e la frase di apertura, — ad esempio "Cari tutti," — seguiti dal corpo del documento, dalla frase di chiusura — ad esempio "Cordialità" —, dalla firma e dall'eventuale lista di allegati.

Si deve osservare che nella prima fase di sperimentazione si è preferito che la sigla dello scrivente dovesse essere fisicamente apposta a penna una volta stampata la lettera. In futuro l'inserimento di un'immagine con una firma digitalizzata è semplice questione di uso del comando `\includegraphics`.

Il logo del Comune di Napoli è posizionato nello spazio della testatina (*heading*). La prima testatina è diversa da quelle che compaiono sulle pagine successive alla prima. Questa specifica del manuale della *corporate identity* è direttamente configurabi-

4. Si veda: <http://www.komascript.de>.

le attraverso l'interfaccia della classe `scr1tr2`, che prevede un'apposita impostazione per le testatine successive alla prima. In ogni caso le testatine sono composte inserendo il nome del 'Dipartimento' e del 'Servizio' scrivente. Queste informazioni sono impostate ad alto livello dall'utente di FACILE attraverso la schermata di configurazione della firma (figura 6). Analoghe informazioni vengono inserite nel piedino.

Sulla prima pagina è mostrato un riquadro per l'apposizione dello *sticker* che reca il codice del protocollo e la data (necessario per tutte le comunicazioni ufficiali destinate all'archiviazione nel sistema documentale dell'Ente). Al momento questa operazione avviene in seguito alla stampa del documento, cioè l'etichetta adesiva con il protocollo non è generata contestualmente alla lettera ma proviene fisicamente da un apposito ufficio. Una estensione futura di questa funzionalità consiste nel reperimento *online* del codice del protocollo, direttamente in fase di finalizzazione del documento in modo che la composizione di questo elemento (codice a barre incluso) avvenga insieme a tutti gli altri attraverso Xe_{La}TeX.

Lo spazio per la firma è lasciato in bianco. Sotto lo spazio per la sigla (che per il momento deve essere apposta a penna sul documento stampato) compare il nome del firmatario e la sua qualifica (ruolo). Una estensione futura prevede l'inserimento di una firma digitale, configurabile ad alto livello attraverso il *front-end*, inserita a basso livello dal motore di composizione.

Data la dimensione del file *template*, di seguito si riporteranno solo alcune tra le soluzioni implementative adottate nel codice.

3.3.2 Qualche dettaglio sul codice sorgente

La parte iniziale del template ha un preambolo strutturato come segue (con evidenziazione separata per i comandi introdotti dalla classe `scr1tr2`):

```
\documentclass[paper=a4,fontsize=12pt,version=last]{scr1tr2}

% impostazioni aggiuntive della classe scr1tr2
\KOMAOPTIONS{enlargefirstpage,twoside=false,
  foldmarks=false,fromalign=left,fromrule=false,
  pagenumber=footright}

% Pacchetti specifici per XeLaTeX
\usepackage{fontspec,xltxtra,xunicode}
\defaultfontfeatures{Scale=MatchLowercase,
  Mapping=tex-text}
\setmainfont[Mapping=tex-text]{
  Garamond 3 LT Std}
\setsansfont[Mapping=tex-text]{
  Frutiger LT 55 Roman}
\usepackage{polyglossia}
\setdefaultlanguage{italian}

% Pacchetti aggiuntivi
\usepackage{calc,xifthen,xspace,url}
\urlstyle{same}
```

```
\usepackage{graphicx,xcolor,varwidth,relsize,
  ragged2e,paralist,marvosym,eurosym,textcomp,
  alphalph,pagesLTS}
```

```
% Alcuni termini italiani (vedi "scrguien.pdf" p. 212)
\providecaptionname{italian}\phoneofficename{
  ufficio}
\providecaptionname{italian}\phonelaboratoryname{
  lab.}
\providecaptionname{italian}\phonemobilename{
  cellulare}
\providecaptionname{italian}\phoneprivatenamename{
  privato}
\providecaptionname{italian}\phoneemergencynamename{
  emergenza}
```

```
% Impostazioni generali riguardanti il layout (vedi "scrguien.
  pdf" p. 185)
```

```
\pagestyle{headings}
% ...
```

Seguono delle impostazioni riguardanti l'aspetto delle testatine e dei piedini:

```
\makeatletter
% larghezza della prima testatina
\@setlength{firstheadwidth}{\paperwidth}
\@addtoplength{firstheadwidth}{-30mm}
% larghezza del primo piedino
\@setlength{firstfootwidth}{\uselength{
  firstheadwidth}}
% posizione della prima testatina
\@setlength{firstheadhpos}{15mm}
\@setlength{firstheadvpos}{15mm}
\addtolength{\textheight}{5mm}
% posizione del primo piedino
\@addtoplength{firstfootvpos}{1mm}
\@setlength{firstfoothpos}{\uselength{
  firstheadhpos}}
% posizione orizzontale della lista
\@setlength{toaddrhpos}{15mm}
% posizione verticale della lista
\@setlength{toaddrvpos}{80mm}
% larghezza della scatola contenente
\@setlength{toaddrwidth}{28.5cm}
% indentazione della lista dei destinatari
\@setlength{toaddrindent}{8.2cm}
% indirizzo di risposta
\@setlength{backaddrheight}{0pt}
% larghezza box informazioni supplementari
\@setlength{locwidth}{5cm}
% posizione orizzontale box informazioni supplementari
\@setlength{lochpos}{%
  21cm-\uselength{locwidth}-\uselength{
  firstheadhpos}}
% posizione verticale box informazioni supplementari
\@setlength{locvpos}{52mm}
% offset orizzontale dell'area-firma
\@setlength{sigindent}{8cm}
% offset verticale dopo la frase di chiusura
\@setlength{sigbeforevskip}{6pt}
\makeatother
```

Successivamente si hanno i comandi che impostano l'area di stampa in alto a sinistra della prima pagina, subito al di sotto dell'intestazione con il logo e il Dipartimento scrivente (si veda la figura 10). Nel codice seguente si osserva il modo in cui si effettua il disegno del contrassegno rappresentante l'area destinata allo *sticker* con il protocollo. Suc-

cessivamente si riporta anche il codice che gestisce l'eventuale campo non nullo contenente la data:

```
% Area—Riferimenti: protocollo, data, ecc.
\removeeffields
% variabile date resa nulla in modo da definirne una locale (
  vedi "myref1") per consentire la personalizzazione dell'
  area—riferimenti
\setkomavar{date}{\null}
\providecaptionname{italian}\datename{}

% MYREF
% (vedi refwidth, refvpos, refhpos in "scrguien.pdf" p. 201)
% Area—protocollo
\providecaptionname{italian}\myrefname{\relsize{2}
  Prot.}
\setkomavar{myref}{%
  \makebox[6.7cm][l]{% vedi refwidth
    % riquadro per l'apposizione dello sticker
    % del protocollo (6.7 x 3.2 cm)
    \mbox{}%
    \color{red}%
    \raisebox{3.2cm}{%
      \rule{2pt}{9pt}\rule{7.0pt}{1em}{2pt}%
    }%
    \mbox{}\hfill\mbox{}%
    \makebox[0pt][r]{%
      \rule{0pt}{3.2cm}\rule{0.0pt}{1em}{2pt}
    }\rule{2pt}{9pt}
  }}
\addtoeffields{myref}

% MYREF1
% per un campo data non vuoto viene stampata la
% stringa di etichettatura "Data"
% per un campo data vuoto la stringa diventa nulla
\newboolean{FCLdatelsEmpty}
\setboolean{FCLdatelsEmpty}{true}
\def\tempDate#1{%
  \ifthenelse{\isempty{#1}}{%
    {}%
    {\relsize{2}Data%
      \setboolean{FCLdatelsEmpty}{false}}
}%
\def\tempDatea{%
  \tempDate{%
    %$field_date%
  }%
}
\newkomavar[\tempDatea]{myref1}
\setkomavar{myref1}{%
  \makebox[4cm][l]{% see refwidth
    %$field_date%
  }%
}
% aggiunge effettivamente la data all'area—riferimenti
\addtoeffields{myref1}
```

Il template non è un sorgente direttamente compilabile. Esso viene utilizzato per produrre un sorgente compilabile dopo che alcune stringhe segnaposto sono state opportunamente sostituite. Le stringhe segnaposto sono state definite secondo il seguente schema:

`%$field_⟨nome-campo⟩`

La parte `⟨nome-campo⟩` corrisponde ad uno dei campi della schermata di FACILE mostrata nella figura 8. Il parser cattura il contenuto testuale del

campo e lo sostituisce alla stringa segnaposto corrispondente, presente nel template. Si consideri, ad esempio, il campo della data. Questo è anche un caso particolare di gestione avanzata della sostituzione. La webform propone all'utente un campo data che non sempre è necessario riempire, perché la data è presente sullo *sticker* del protocollo che verrà applicato una volta stampata la lettera. Quindi è necessario un algoritmo che controlli se la stringa della data catturata dal parser è vuota. In caso positivo lo spazio della lettera riservato alla data rimane completamente bianco. Nel caso in cui l'utente immette una data il campo `%$field_date` potrà essere sostituito con la stringa corrispondente — nell'esempio della figura 9 la stringa è "31 agosto 2011". Nel codice precedente questo algoritmo è implementato nelle macro `\tempDate` e `\tempDatea`. Nell'esempio citato l'utente ha impostato una data non vuota e in fase di composizione viene stampata la stringa "Data" in una posizione opportuna del foglio e sotto di essa la stringa "31 agosto 2011" (si veda la figura 10).

Proseguendo a esaminare il preambolo del file *template*, nel codice seguente si imposta il posizionamento di alcuni elementi aggiuntivi:

```
% posizione e dimensioni dell'area riferimenti
\makeatletter
%
\@setlength{refhpos}{%
  0.7\useplength{firsttheadhpos}
}
\@setlength{refvpos}{70mm}
\@setlength{refwidth}{50mm}
\@setlength{refaftervskip}{2cm}
%
\makeatother

Seguono altre impostazioni:

% Dipartimento
\newkomavar{DeptName}
\setkomavar{DeptName}{%
  %$field_department_name%
  \mbox{}%
}
% Servizio
\newkomavar{ServName}
\setkomavar{ServName}{%
  %$field_service_name%
  \mbox{}%
}

% Mittente, (va nell'area—firma)
\setkomavar{fromname}{%
  %$field_signature%
  \mbox{}
}

% Area—firma
\setkomavar{signature}{%
  %$field_signature%
  \mbox{}
  \\
  \textit{%
    %$field_role_signature%
  }%
}
```

```
% Testatina in prima pagina (First page header)
\makeatletter
\firsthead{%
  \begin{tabular}{@{}l@{}}
    \makebox[0.0\textwidth][l]{%
      \includegraphics[%
        width=3.5cm%
      ]{%
        LOGO_COMUNE_NAPOLI.png%
      }%
    }%
  %%
  \parbox{1.0\useplength{firstheadwidth}}{%
    \raggedright
    \sffamily\bfseries\normalsize%
    \usekomavar{DeptName}\\
    {%
      \relsize{0}%
      \usekomavar{ServName}%
    }%
  }%
}
\end{tabular}
}
\setlength{fromrulethickness}{0pt}
\makeatother

% PIEDINI e FIRMA, macro con gestione di campi vuoti
\newcommand{\FCLaddressSignature}[1]{%
  \ifthenelse{\isempty{#1}}{%
    {}% if #1 is empty
  }{%
    {#1}% if #1 is not empty
  }%
}
\newcommand{\FCLtelSignature}[1]{%
  \ifthenelse{\isempty{#1}}{%
    {}% if #1 is empty
  }{%
    {$\cdots$ \bfseries tel.} #1}% if #1 is not empty
  }%
}
\newcommand{\FCLfaxSignature}[1]{%
  \ifthenelse{\isempty{#1}}{%
    {}% if #1 is empty
  }{%
    {$\cdots$ \bfseries fax} #1}% if #1 is not empty
  }%
}
\newcommand{\FCLemailSignature}[1]{%
  \ifthenelse{\isempty{#1}}{%
    {}% if #1 is empty
  }{%
    \bfseries\url{#1}}% if #1 is not empty
  }%
}
\newcommand{\FCLurlSignature}[1]{%
  \ifthenelse{\isempty{#1}}{%
    {}% if #1 is empty
  }{%
    {$\cdots$ \bfseries\url{#1}}}% if #1 is not empty
  }%
}

% Piedino della prima pagina (First footer)
\makeatletter
\firstfoot{%
  \parbox{16.51cm}{% magic number, larghezza box
    contenente footer a cui si affianca il numero di pagina
    \normalfont\footnotesize\itshape
    \FCLaddressSignature{%
      %$field_address_signature%
    }%
    \xspace
    \FCLtelSignature{%
      %$field_tel_signature%
    }%
    \xspace
    \FCLfaxSignature{%
      %$field_fax_signature%
    }%
    \xspace
    \FCLemailSignature{%

```

```
%$field_email_signature%
  }\xspace
  \FCLurlSignature{%
    %$field_web_signature%
  }
}% ... end of parbox
% Numero di pagina
\ifnum\value{pagesLTS.pagenr}>1%
\makebox[0pt][l]{% magic numbers, aggiustamenti
verticali
  \raisebox{3pt}[0pt][0pt]{%
    \rule[3pt-\f@baselineskip]{0.8pt}{8pt+\f@baselineskip}%
    \rule[3pt]{0pt}{% spacer
    \relsize{-1}\sffamily\upshape
    \pagemark di \pageref{LastPage}
  }% ... raisebox
}% ...makebox
\fi
}
\makeatother
```

Il codice precedente introduce delle macro come \FCLaddressSignature, \FCLtelSignature, \FCLfaxSignature, eccetera, che gestiscono dei campi lasciati eventualmente in bianco dall'utente nella definizione della firma. I campi nulli non vengono riportati nel piedino, la cui composizione procede senza errori in tutti i casi d'uso.

Infine, si riporta un estratto del resto del codice:

```
% gestione margini, ecc.
%
\usepackage[left=3cm,right=3cm,top=8cm,bottom=3cm]
{geometry}
\setlength{\footskip}{1.0cm}

\begin{document}
%
\pagenumbering{arabic}% richiesto dal pacchetto
pagesLTS

\FCLcheckCcAddressListEmpty{%
  % assegna \boolean{FCLccAddressListIsEmpty}
  %$field_address_cc%
}%
% ...

\setkomavar{subject}{%
  % controlla se va spezzata la lista
  \ifx\FCLaddressListSplitted\undefined\relax
    \ifdim\EZTsizePopBoxa >4.6cm
      \rule[0pt]{\FCLsizePopBoxa-5.5cm}{%
        par% riga distanziatrice verticale
        \mbox{}%
      }%
    \fi
  \else
    % lista spezzata
    \mbox{}%
    %
    \newpage%..... salto pagina
    % ...
    % ...
  \fi
  % ORA COMPONE L'OGGETTO
  \noindent
  \mbox{%
    {\sffamily Oggetto:}\ \parbox[t]{%
      textwidth-1.7cm\relax}{%
        \noindent
```

```

\hyphenpenalty=5000
\tolerance=1000
%$field_object%
}
}
}

\begin{letter}{
% ...
% FORMULA DI APERTURA
\FCLcheckOpeningEmpty{%
%$field_opening%
}

\opening{%
%
%$field_opening%
\mbox{}\ifthenelse{\boolean{FCLopeningsIsEmpty}
}{\{,\}}%
}

% -----
% BEGIN LETTER BODY
%
\noindent
%
%$field_body%
%
% -----
% END LETTER BODY
%
% CLOSING PHRASE
%
\FCLcheckClosingEmpty{%
%$field_closing_phrase%
}%

\closing{%
\enlargethispage*{2.0\baselineskip}% starred
version tries to squeeze material in page
\mbox{}
%
%$field_closing_phrase%
%
\ifthenelse{\boolean{FCLclosingsIsEmpty}}{\{,\}}%
\mbox{}
}
%
% ALLEGATI
\FCLcheckAttachmentsListEmpty{%
%$field_attachments%
}
%
%
\ifthenelse{\boolean{FCLattachmentsListIsEmpty}}{\{,\}}%
%
%
% Allegati presenti
%
\noindent\mbox{}\rule{0pt}{3.0\baselineskip}%
spacer
\\
\noindent{\sffamily\bfseries Allegati:}\\
\begin{compactenum}[\mbox{\itshape Allegato\ } A.]
%
\pprowsAttachments{\{,%
%$field_attachments
}
\end{compactenum}
%
}% ...endif list is empty

```

```

%
\end{letter}
\end{document}

```

4 Conclusioni

È stata presentata la *web application* FACILE, un'applicazione *server-based* che utilizza la distribuzione T_EX Live per comporre comunicazioni ufficiali ad uso interno dell'Amministrazione comunale di Napoli. Le lettere prodotte con questo strumento sono conformi alle specifiche del Manuale della *Corporate Identity* del Comune di Napoli. Si tratta probabilmente del primo esperimento in Italia in cui si utilizza L^AT_EX a questo livello in una pubblica amministrazione.

Ringraziamenti

Si ringrazia il Comune di Napoli, la Engineering Ingegneria Informatica S.p.a. e la Kelyon S.r.l. per il supporto tecnico offerto durante la stesura di questo articolo. Si ringrazia inoltre Enrico Gregorio per i suggerimenti sull'implementazione di alcune macro, forniti attraverso il forum di G_JT.

Riferimenti bibliografici

- ANONIMO (2006). «Manuale della Corporate Identity del Comune di Napoli». Comune di Napoli, Dipartimento Comunicazione Istituzionale e Immagine, <http://www.comune.napoli.it>.
- KOHM, M. e MORAWSKI, J.-U. (2011). «KOMAScript a versatile L^AT_EX 2_ε bundle». Disponibile su www.ctan.org.

- ▷ Agostino De Marco
Università degli Studi di Napoli
“Federico II”
Dipartimento di Ingegneria
Aerospaziale
Via Claudio 21
80125 Napoli
agostino dot demarco at unina
dot it
- ▷ Paolo Eugenio Cresci
Dirigente Informatico
del Comune di Napoli
Servizio SIAD – Sistema Informativo
Amministrativo e Documentale
Piazza Giovanni XXIII 6
80126 Napoli
paoloeugenio dot cresci at
comune dot napoli dot it