

Il markup di documenti (I parte)

Fabio Vitali



«Quando io uso una parola», disse Humpty Dumpty in tono piuttosto sprezzante, «significa quello che io scelgo che significhi - né più, né meno.»

Lewis Carroll
“Attraverso lo specchio”

Introduzione

Qui esaminiamo:

- ◆ Cos'è il markup
- ◆ Che tipi di markup esistono
- ◆ Caratteristiche di markup procedurale e descrittivo
- ◆ Una brevissima storia del markup su computer

Perché tutto questo chiasso?

- Quando si porta una collezione di documenti in forma elettronica, si ha di solito in mente in generale una specifica applicazione (metterla in rete, prepararla per la stampa, ecc.).
- Di solito, si cerca di trasformare il documento nella forma più opportuna perché venga utilizzato nell'applicazione suddetta.
- Spesso per questo si fanno delle scelte che impediscono o ostacolano notevolmente un ulteriore riuso della stessa collezione per una diversa applicazione. L'impaginato poco si adatta ad un'indicizzazione per la rete, o viceversa i linguaggi di visualizzazione su rete sono troppo poco sofisticati per una produzione tipografica di buon livello, ecc.
- I linguaggi di markup derivati da SGML sono i linguaggi più opportuni per strutturare e marcare i documenti in maniera indipendente dall'applicazione, favorendo la **riusabilità**, la **flessibilità** e la **apertura ad applicazioni complesse**.

Due problemi

Rendere i contenuti accessibili a chiunque

- ◆ Gestire in maniera semplice e uniforme tutte le fasi di gestazione di un prodotto editoriale
- ◆ Gestire in maniera semplice ed uniforme tutti i possibili usi di un medesimo contenuto: su carta, su video, su terminale Braille, su sintetizzatore vocale, ecc.
- ◆ Gestire in maniera semplice ed uniforme tutti i possibili riusi di un medesimo contenuto: libro, indice, sito web, catalogo, archivio, ...

Rendere il testo accessibile nel tempo

- ◆ Il recupero dei dati del Census Bureau del 1960 (1976-1982)
- ◆ Il recupero dei dati per il progetto LUNR del 1969 (1986-????)
- ◆ Brewster Kahle <http://www.archive.org/>
- ◆ Bruce Sterling <http://www.deadmedia.org/>
- ◆ Boeing e la digitalizzazione dei manuali degli aerei.

Obsolescenza dei documenti

I vecchi documenti elettronici non sono più accessibili...

- ... quando realizzati con applicazioni *ad hoc* e senza documentazione
- ... quando realizzati con applicazioni commerciali obsolete
- ... quando realizzati per periferiche obsolete
- ... quando memorizzati su media obsoleti

Accessibilità nel tempo (2)

Problema HW: migrazione dei dati

- ◆ La predominanza degli hard disk sui medium esterni e il costo in diminuzione dell'hardware connesso rendono possibile il trasferimento di tutti i dati sulla nuova macchina appena acquistata

Problema SW: conversione

- ◆ Non altrettanto semplice, non altrettanto efficace, non altrettanto automatizzabile, non altrettanto universale.
- ◆ Richiede ancora una buona dose di lavoro manuale, in quantità proporzionale alla quantità di dati da trattare.

Entrambe richiedono attenzione costante nel tempo, ma per il software è più difficile.

Problema software

Per i documenti del passato:

- ◆ Chi lo sa? Ci sono molte iniziative
- ◆ Più aspettiamo, peggio è

Per i documenti del presente e del futuro:

- ◆ Evitiamo di ripetere le ingenuità del passato
- ◆ Prepariamo la transizione all'obsolescenza
- ◆ Sfruttiamo la rete per il riversamento
- ◆ Sfruttiamo gli standard per i formati dei dati

La fonte dei problemi

Formati binari

- ◆ Il formato dati è un'immediata trasposizione del dato nella memoria. Nessun elemento del documento ha un senso comprensibile senza l'applicazione che l'ha generato.

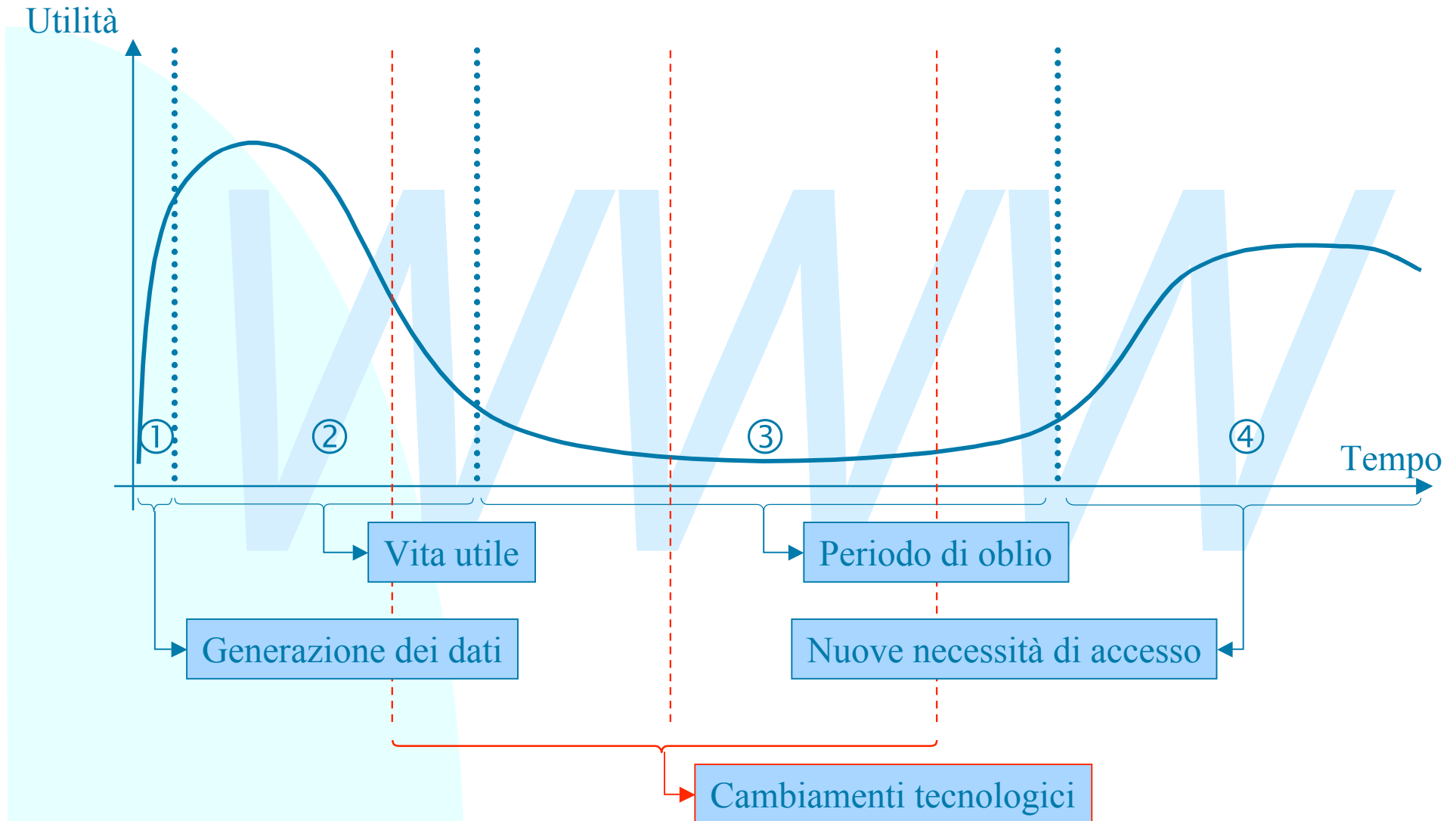
Formati proprietari

- ◆ L'azienda produttrice dell'applicazione decide come è fatto il documento secondo logiche interne al prodotto

Formati orientati all'applicazione

- ◆ Il contenuto del documento è inframmezzato da istruzioni di formattazione specifiche, che rendono difficile il riutilizzo del documento per altri scopi a parte quello per cui è stato creato.

L'utilità dei dati digitali nel tempo



A seguire: Alcuni approcci alla soluzione

Alcuni approcci alla soluzione

Stampa su carta

- ◆ Possibile fino a tardi (2, inizio di 3); Accesso ma non riuso

Preservazione delle tecnologie

- ◆ Costoso. Necessario agire continuamente (2, 3, 4). Accesso ma non riuso

Emulazione delle tecnologie

- ◆ Costoso. Possibile in fase 4 se documentazione sufficiente. Accesso e riuso parziale.

Migrazione a nuove tecnologie

- ◆ Necessario agire continuamente. Accesso e riuso

Incapsulamento

- ◆ Entro la fine della fase 2. Accesso e riuso

Adozione di formati ricchi

- ◆ Entro la fase 1. Poco costosa. Accesso e riuso.

A seguire: Formati dati proprietari

Formati dati proprietari

Word Processor

- ◆ MS Word, Word Perfect, Wordstar, XY Write, etc.
- ◆ Tutti differenti, rapida obsolescenza, decente compatibilità verso il passato

Desktop Publishing

- ◆ PageMaker, FrameMaker, Quark XPress, Ventura Publisher, etc.
- ◆ Tutti differenti, tutti estremamente specializzati, rapidissima obsolescenza, nessuna compatibilità col passato

DBMS

- ◆ Oracle, MS Access, MS SQLserver, IBM DB2, etc.
- ◆ Estremamente orientati all'efficienza, opacissimi, decente obsolescenza, buona compatibilità col passato

Presentazione

- ◆ Postscript, PDF, ma anche Flash, Shockwave
- ◆ Opachi, specializzati, complessi. Compatibilità col passato?

Livelli diversi di riuso

- Riaccedere in lettura ai dati
 - ◆ Corretta lettura del medium fisico
- Reinterpretare i dati
 - ◆ Corretta identificazione delle caratteristiche del formato dati
- Rieseguire le applicazioni di manipolazione
 - ◆ Disponibilità e compatibilità (diretta o via emulazione) delle applicazioni originali
- Modificare e aggiornare i dati
 - ◆ Conversione dei dati alle applicazioni odierne senza perdita di informazioni
- Realizzare nuove funzionalità sui dati
 - ◆ Adattare le vecchie informazioni alle nuove applicazioni in modo da permettere l'esecuzione di nuove funzionalità

Cos'è il markup? (1)

- Definiamo markup *ogni mezzo per rendere esplicita una particolare interpretazione di un testo.*
- Per esempio, tutte quelle aggiunte al testo scritto che permettono di renderlo più fruibile.
- Oltre a rendere il testo più leggibile, il markup permette anche di specificare ulteriori usi del testo.
- Con il markup per sistemi informatici (il nostro caso), specifichiamo le modalità esatte di utilizzo del testo nel sistema stesso.
- Il markup non è soltanto un inevitabile e sgradevole risultato della informatizzazione dell'arte tipografica. Non è qualcosa che sta con noi a causa dell'informatica.

Cos'è il markup? (2)

- Quando un autore scrive, da millenni a questa parte, specifica anche i delimitatori di parola (chiamati *spazi*), i delimitatori di frase (chiamati virgole) e i delimitatori di periodo (chiamati punti).
- La numerazione delle pagine o l'uso dei margini per creare effetti sul contenuto sono noti da centinaia di anni.
- Eppure questo a stretto rigore non fa parte del testo, ma del markup: nessuno dirà ad alta voce 'virgola' o 'punto' nel leggere un testo, ma creerà adeguati comportamenti paralinguistici (espressioni, toni, pause) per migliorare in chi ascolta la comprensione del testo.

Modi del markup: proprietario vs. pubblico

Un formato proprietario è stato creato da una specifica azienda con uno specifico scopo commerciale.

L'azienda ne detiene i diritti, e dunque è in grado di modificarlo, aggiornarlo o rivoluzionarlo in qualunque momento e per qualunque motivo.

Un formato pubblico è stato creato da un gruppo di interesse (individui, aziende, enti non commerciali, ecc.) come modello di armonizzazione tra le esigenze di ciascun partecipante.

Il gruppo tipicamente pubblica le specifiche del formato, permettendo a chiunque di realizzare strumenti software per quel formato. A volte questo si concretizza in uno standard ufficiale, avente valore normativo.

A seguire: Modi del markup:

Modi del markup: binario vs. leggibile

Un formato binario è la memorizzazione esatta delle strutture in memoria dell'applicazione, che niente hanno a che vedere con le esigenze di comprensione di esseri umani. Il testo non è visibile o è visibile per caso.

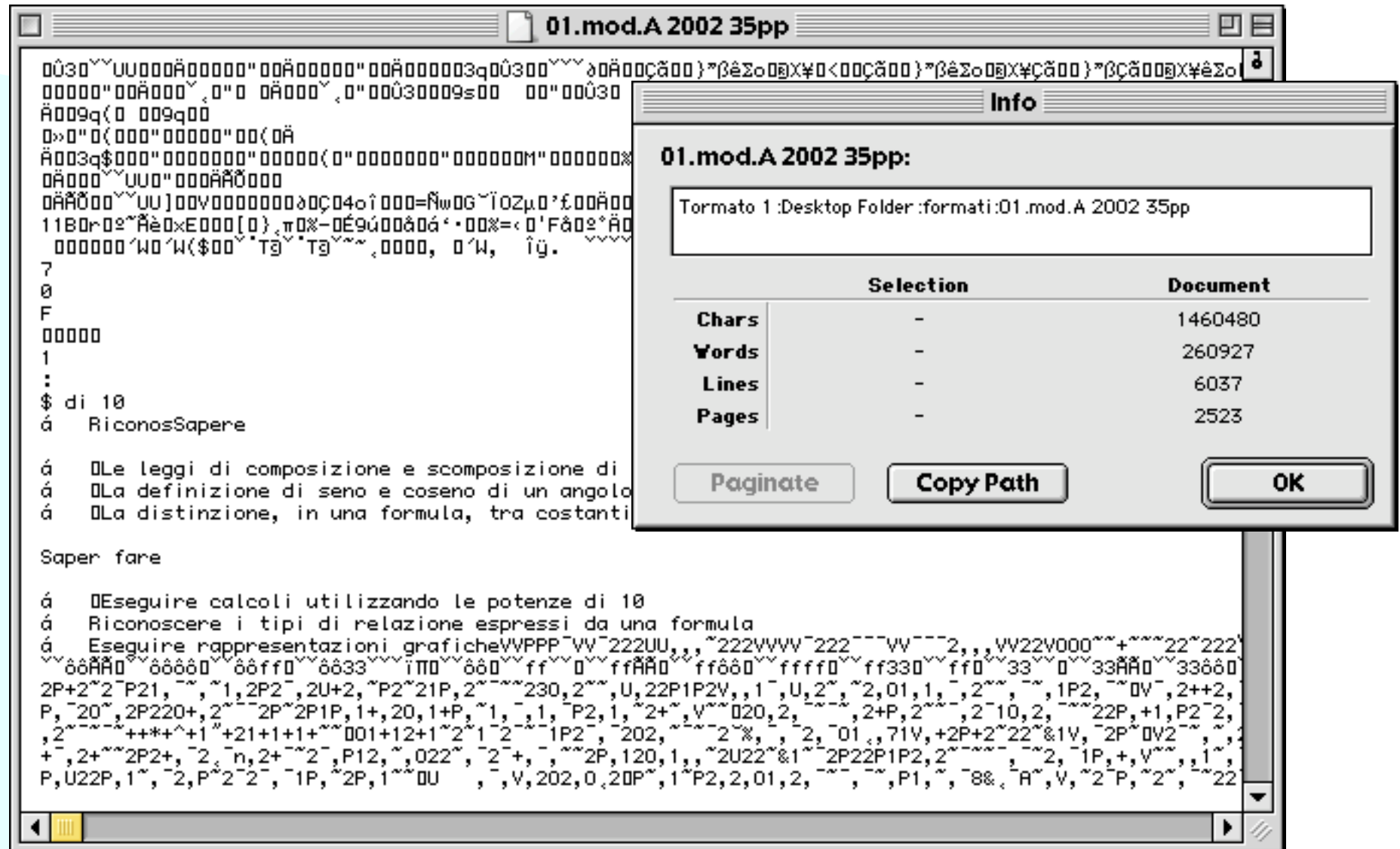
Un formato leggibile invece è fatto per essere, in casi speciali, letto anche da esseri umani, che possono intervenire per operazioni di emergenza.

L'applicazione deve trasformare quanto legge in una struttura interna utile per le operazioni di modifica o presentazione. Questa fase si chiama *parsing*.

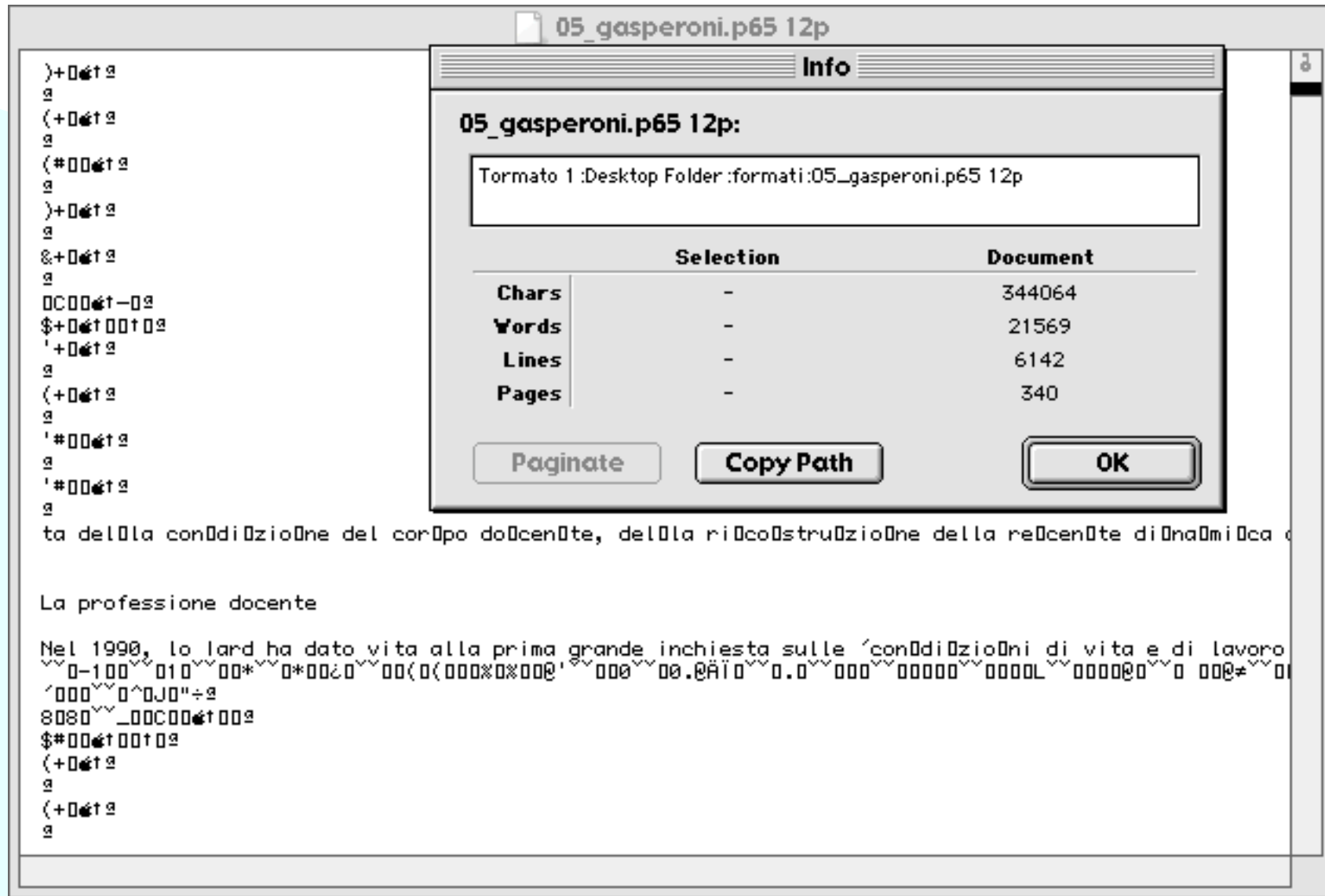
Esempio: .doc di MS Word



Esempio: Quark Xpress



Esempio: Adobe PageMaker



Esempio: PDF



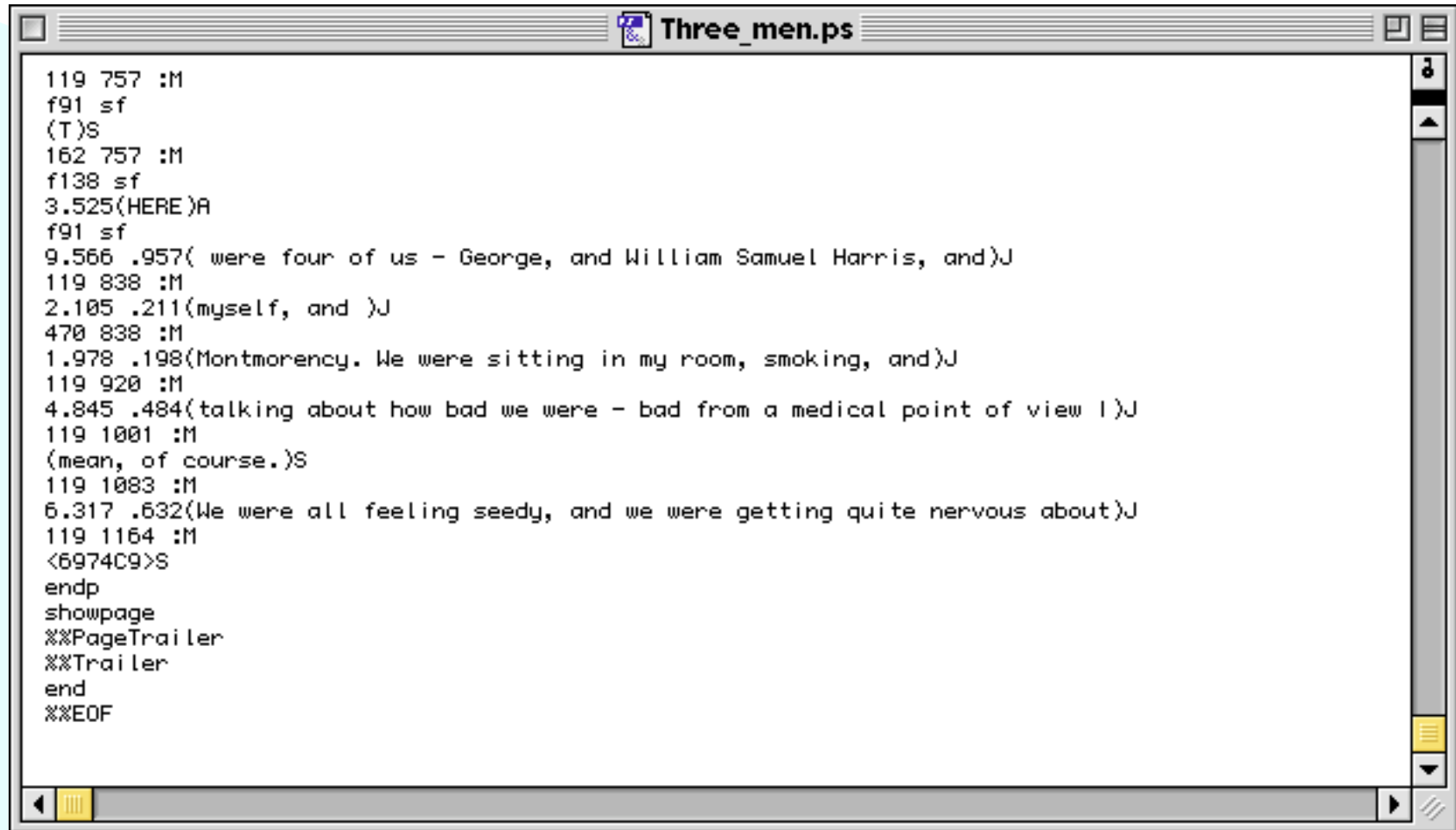
```

%PDF-1.2
%  

4 0 obj
<<
/Type /Page
/Parent 5 0 R
/Resources <<
/ProcSet 2 0 R
>>
>>
endobj
9 0 obj
<<
/Length 10 0 R
/Filter /FlateDecode
>>
stream
DHâ}...n@000†' -; *ΣÜφr0jπ0m0ÈUË>mi8)b0éÁto_lq0)j, ^δz05[Δ2%[m≠áh#×0
i#0)Á^μ(†#Ω0◊πy/h00y
^Avçt$05^#Á"πx4ö]π00"GA5k
z0-îH), °-0[_íπév-#*â'æ30, *#Eá% ¥00ú0a+\òzâ0CDSú d*~0/ÁìΔ#uR^Y5ò02è0*e=f
9mΣ...>·0)`ùç%JTá2úò/0JòNLOç≤†0+K; iÜ?}0RLø: Ááπ0E0y0o0E2\fi, \NΣæ*k0ú0ä0**/ÿ0í 10è20(5~i{ÿ™·iÈ?δ÷|7-, "0
0ò00%CHOÿ|]æ?b#0Á_èÜ"9l(UÜ-ÿ0'05X"00fáwó'0^òÿ/fG+Æ≤(vδ÷e<l·y'◊"0*, )æEí0úuRΔ:È?i0^,
endstream
endobj
10 0 obj
691
endobj
6 0 obj
<<
/Type /Page
/Parent 5 0 R

```


Esempio: PostScript

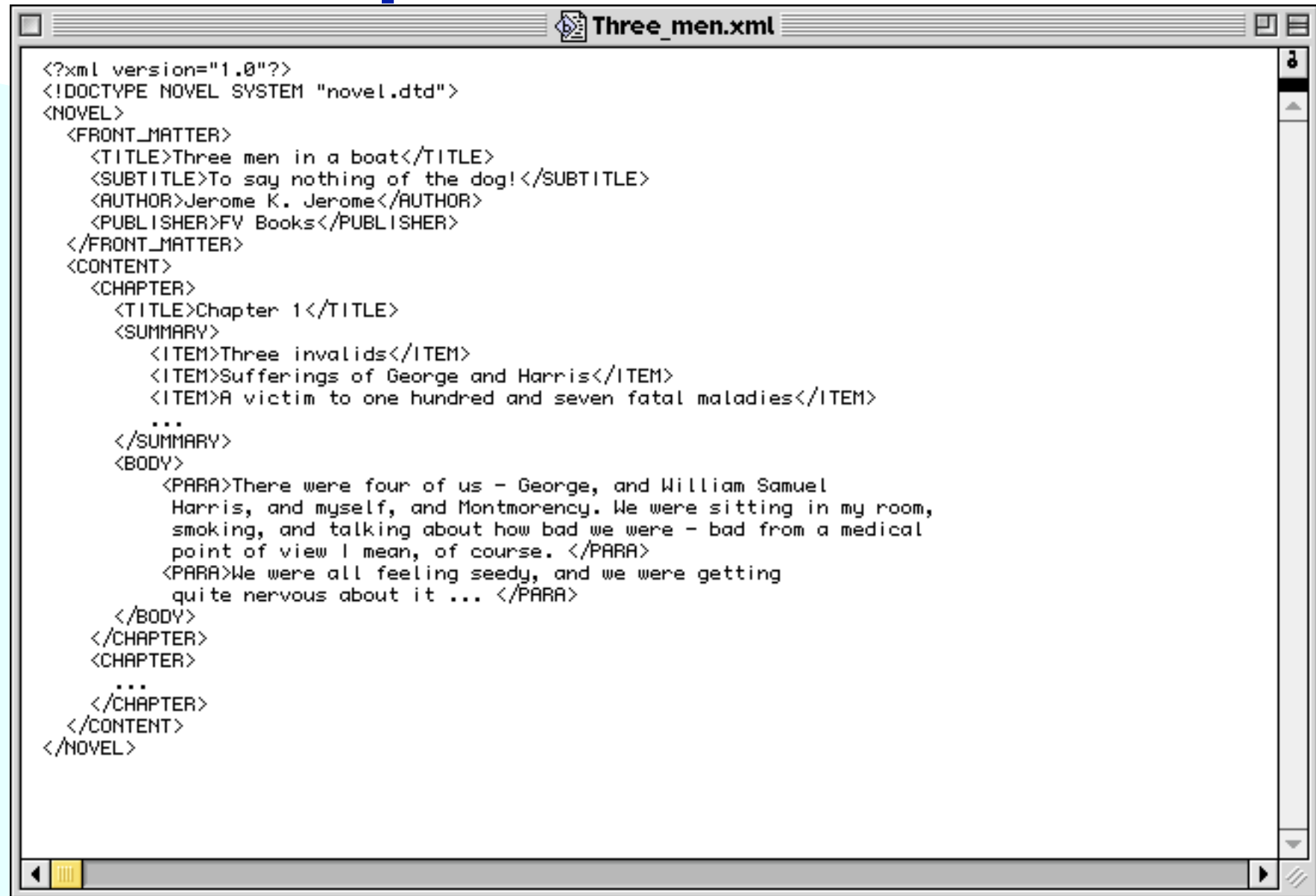


```
119 757 :M
f91 sf
(T)S
162 757 :M
f138 sf
3.525(HERE)A
f91 sf
9.566 .957( were four of us - George, and William Samuel Harris, and)J
119 838 :M
2.105 .211(myself, and )J
470 838 :M
1.978 .198(Montmorency. We were sitting in my room, smoking, and)J
119 920 :M
4.845 .484(talking about how bad we were - bad from a medical point of view I)J
119 1001 :M
(mean, of course.)S
119 1083 :M
6.317 .632(We were all feeling seedy, and we were getting quite nervous about)J
119 1164 :M
<6974C9>S
endp
showpage
%%PageTrailer
%%Trailer
end
%%EOF
```

Esempio: HTML

```
Three_men.html
<HTML>
<HEAD>
  <META HTTP-EQUIV="Content-Type" CONTENT="text/html; charset=iso-8859-1">
  <TITLE>Three men in a boat</TITLE>
</HEAD>
<BODY>
  <P ALIGN="CENTER">&nbsp;</P>
  <FONT SIZE=7>
    <P ALIGN="CENTER">Three men in a boat</P>
  </FONT>
  <FONT SIZE=6>
    <P ALIGN="CENTER">To say nothing of the dog!</P>
    <P ALIGN="CENTER">&nbsp;</P>
    <P ALIGN="CENTER">Jerome K. Jerome</P>
    <P ALIGN="CENTER">&nbsp;</P>
    <P ALIGN="CENTER">FV Books</P>
  </FONT>
  <P ALIGN="CENTER">&nbsp;</P>
  <FONT SIZE=5>
    <P ALIGN="CENTER">Chapter 1</P>
  </FONT>
  <DIR><DIR>
    <FONT SIZE=5><I>
      <P>Three invalids - Sufferings of George and Harris -
      A victim to one hundred and seven fatal maladies - &#133; </P>
    </I></FONT>
  </DIR></DIR>
  <FONT SIZE=5>
    <P ALIGN="JUSTIFY">There were four of us - George, and William Samuel
    Harris, and myself, and Montmorency. We were sitting in my room,
    smoking, and talking about how bad we were - bad from a medical
    point of view I mean, of course. </P>
    <P ALIGN="JUSTIFY">We were all feeling seedy, and we were getting
    quite nervous about it&#133;</P>
  </FONT>
</BODY>
</HTML>
```


Esempio: XML



```
<?xml version="1.0"?>
<!DOCTYPE NOVEL SYSTEM "novel.dtd">
<NOVEL>
  <FRONT_MATTER>
    <TITLE>Three men in a boat</TITLE>
    <SUBTITLE>To say nothing of the dog!</SUBTITLE>
    <AUTHOR>Jerome K. Jerome</AUTHOR>
    <PUBLISHER>FV Books</PUBLISHER>
  </FRONT_MATTER>
  <CONTENT>
    <CHAPTER>
      <TITLE>Chapter 1</TITLE>
      <SUMMARY>
        <ITEM>Three invalids</ITEM>
        <ITEM>Sufferings of George and Harris</ITEM>
        <ITEM>A victim to one hundred and seven fatal maladies</ITEM>
        ...
      </SUMMARY>
      <BODY>
        <PARA>There were four of us - George, and William Samuel
        Harris, and myself, and Montmorency. We were sitting in my room,
        smoking, and talking about how bad we were - bad from a medical
        point of view I mean, of course. </PARA>
        <PARA>We were all feeling seedy, and we were getting
        quite nervous about it ... </PARA>
      </BODY>
    </CHAPTER>
    <CHAPTER>
      ...
    </CHAPTER>
  </CONTENT>
</NOVEL>
```

Modi del markup: interno vs. esterno

Il markup interno inserisce istruzioni di presentazione all'interno del testo, in mezzo alle parole.

Il markup esterno prevede due blocchi di informazioni: il contenuto e il markup, separati e collegati da meccanismi di indirazione

Il markup interno richiede sintassi particolari per distinguere il markup dal contenuto. Tipicamente si adottano segnalatori particolari che cambiano il tipo di interpretazione del documento. La presenza del carattere segnalatore nel testo richiede l'adozione di tecniche di *escaping*.

Il markup esterno richiede un meccanismo di indirazione, basato su indirizzi, *offset* o identificatori, per associare con correttezza il markup al contenuto.

Modi del markup: procedurale vs. descrittivo

Il markup assolve a diversi ruoli a seconda del sistema di elaborazione, dell'applicazione, dello scopo a cui il documento è soggetto.

- ◆ Puntuazionale
- ◆ Presentazionale
- ◆ Procedurale
- ◆ Descrittivo
- ◆ Referenziale
- ◆ Metamarkup

Markup puntuazionale

- ◆ Il markup puntuazionale consiste nell'usare un insieme prefissato di segni per fornire informazioni perlopiù sintattiche sul testo.
- ◆ Le regole di punteggiatura sono sostanzialmente stabili, note agli autori, e frequenti nei documenti. Per questo gli autori tipicamente forniscono il loro markup puntuazionale autonomamente.
- ◆ Esistono tuttavia notevoli problemi nell'uso della punteggiatura:
 - ✦ Incertezze strutturali (virgola, punto e virgola o punto?),
 - ✦ Incertezze grafiche (virgolette aperte e chiuse o neutre?),
 - ✦ ambiguità procedurali (il punto viene usato sia per segnare la fine di una frase, che l'esistenza di un'abbreviazione, senza contare i tre puntini di sospensione).

Markup presentazionale

- ◆ Il markup presentazionale consiste nell'indicare effetti (grafici o altro) per rendere più chiara la presentazione del contenuto.
- ◆ Nel testo, possono essere cambi di paragrafo o di pagina, interlinea, pallini per liste, ecc.
- ◆ E' altresì markup presentazionale: cambiare pagina all'inizio di una nuova sezione, scrivere "Capitolo 3" in cima alla pagina, ecc.

Markup procedurale

- ◆ Il markup procedurale consiste nell'indicare con precisione ad un sistema automatico che effetto attivare e che procedura (serie di istruzioni) eseguire nella visualizzazione del contenuto.
- ◆ In definitiva, utilizzo le capacità del sistema di presentazione per avere con precisione l'effetto voluto.
- ◆ Esempio: *Wordstar Dot Commands*
 - .PL 66
 - .MT 6
 - .MB 9
 - .LH 12

Markup descrittivo

- ◆ Il markup descrittivo consiste nell'identificare strutturalmente il tipo di ogni elemento del contenuto.
- ◆ Invece di specificare effetti grafici come l'allineamento o l'interlinea, ne individuo il ruolo all'interno del documento, specificando che un elemento è un titolo, un paragrafo, o una citazione.

Markup referenziale

- ◆ Il markup referenziale consiste nel fare riferimento ad entità esterne al documento per fornire significato o effetto grafico ad elementi del documento. Per esempio, utilizzare una sigla nota che venga poi sostituita dalla parola intera durante la stampa
- ◆ Es.: l'autore scrive "CdL" e il sistema trasforma automaticamente l'input in "Corso di Laurea"

Metamarkup

- ◆ Il metamarkup consiste nel fornire regole di interpretazione del markup e permette di estendere o controllare il significato del markup.
- ◆ Ad esempio, la possibilità di definire macro per l'interpretazione e la visualizzazione del documento, o il processo di definizione degli elementi e delle procedure valide di un documento.

Un testo su carta

Tre Uomini in Barca

Jerome K. Jerome

1889

Capitolo primo

Tre invalidi - Le sofferenze di George e Harris - La vittima di centosette malattie inguaribili - [...]

Eravamo in quattro: George, William Samuel Harris, e io, Montmorency. Standocene seduti in camera mia, fumavamo e parlavamo di quanto fossimo malridotti... malridotti, dal punto di vista della salute, intendo, naturalmente. Ci sentivamo tutti piuttosto giù di corda, ...

3

Il testo senza markup

Questo è il testo completamente senza markup, come poteva essere scritto su un papiro della biblioteca di Alessandria, nel II o III secolo a.C.

Treuominiinbarcajeromekjerome1889capitoloprimot
reinvalidilesofferenzedigeorgeeharrislavittimadicent
osettemalattieinguaribilieravamoinquattrogeorge,wil
liamsamuelharriseiomontmorencystandocenesedutii
ncameramiafumavamoeparlavamodiquantofossimo
malridottimalridottidalpuntodivistadellasaluteintend
onaturalmentecisentivamotutti piuttostogiùdicorda

Markup metabolizzato

Aggiungiamo markup puntuazionale e presentazionale: maiuscole/minuscole, punteggiatura, spazi e ritorni a capo sono essi stessi elementi di markup.

Tre Uomini in Barca

Jerome K. Jerome (1889)

Capitolo primo

Tre invalidi - Le sofferenze di George e Harris - La vittima di centosette malattie inguaribili - [...]

Eravamo in quattro: George, William Samuel Harris, e io, Montmorency. Standocene seduti in camera mia, fumavamo e parlavamo di quanto fossimo malridotti... malridotti, dal punto di vista della salute, intendo, naturalmente.

Ci sentivamo tutti piuttosto giù di corda, ...

Markup procedurale

Sono comandi, o istruzioni che il sistema di lettura (umano o elettronico) deve eseguire sul testo. Ad esempio, istruzioni su come andare a capo, come decidere i margini, ecc. Questo è RTF.

```
{\rtf1 \mac \ansicpg10000 \uc1 \pard \plain \s15 \qc \widctlpar \adjustright \f4 \fs48
\cgrid {Tre Uomini in Barca \par } \pard \plain \widctlpar \adjustright \f4 \cgrid {
\line \par \par \par \par \par } \pard \plain \s1 \qc \keepn \widctlpar \outlinelevel0
\adjustright \i \f4 \fs36 \cgrid {Jerome K. Jerome \par } \pard \plain \qc \widctlpar
\adjustright \f4 \cgrid { \fs36 [...] \par 1889 \line \par } \pard \widctlpar \adjustright
{ \page } { \b \fs36 Capitolo primo } { \par \par \par \line } { \i Tre invalidi - Le
sofferenze di George e Harris - La vittima di centosette malattie inguaribili - [
\u8230 \c9] \par } { \line } { \fs28 Eravamo in quattro: George, William Samuel
Harris, e io, Montmorency. Standocene seduti in camera mia, fumavamo e
parlavamo di quanto fossimo malridotti \u8230 \c9 malridotti, dal punto di
vista della salute, intendo, naturalmente. \line Ci sentivamo tutti piuttosto gi
\u249 \9d di corda, ... \par }}
```

Markup descrittivo

Sono informazioni (descrizioni) sugli elementi del documento, che ne specificano il ruolo, la giustificazione, la relazione con gli altri elementi.

```
<ROMANZO><TITOLO>Tre Uomini in Barca</TITOLO>  
<AUTORE>Jerome K. Jerome</AUTORE>  
<ANNO>1889</ANNO>  
<CAPITOLO><TITOLO>Capitolo primo</TITOLO>  
<INDICE><EL>Tre invalidi</EL><EL>Le sofferenze di George e Harris  
</EL><EL> La vittima di centosette malattie inguaribili </EL>[...]</INDICE>  
<PARA>Eravamo in quattro: George, William Samuel Harris, e io,  
Montmorency. Standocene seduti in camera mia, fumavamo e parlavamo di  
quanto fossimo malridotti... malridotti, dal punto di vista della salute, intendo,  
naturalmente. </PARA>  
<PARA>Ci sentivamo tutti piuttosto giù di corda, ...</PARA>  
</CAPITOLO>... </ROMANZO>
```

Il markup procedurale (1)

- Basato sull'aspetto
 - ◆ Ad ogni elemento del documento viene associata la procedura per visualizzarlo in maniera voluta: font, dimensione, corsivi, grassetto, margini, interlinea, ecc.
- Dipendente dal sistema
 - ◆ ogni sistema di visualizzazione impone le proprie regole e la propria sintassi, dipendendo da:
 - ◆ Filosofia sintattica (comandi-punto per troff, comandi-barra per RTF, ecc.)
 - ◆ Capacità di raggruppamento (parentesi graffe in RTF, comando di disattivazione esplicita in troff, ecc.)
 - ◆ Supporto di specifiche funzionalità

Il markup procedurale (2)

- Associato agli individui

- ◆ ogni elemento possiede le proprie procedure per la visualizzazione, che possono anche essere tutte diverse anche per elementi dello stesso tipo.

- Non contestuale

- ◆ Le regole di visualizzazione non dipendono dal contesto in cui vengono fatte, ma ognuna fa specie a sé.

- ◆ Ad esempio, una lista in **troff** è fatta come segue:

```
.li  
    .it elemento 1  
    .it elemento 2  
.el
```

- ◆ Nessun controllo impone di chiudere la lista alla fine, o di usare i comandi `.it` solo dentro alla lista.
- ◆ Inoltre, e molto importante, non è possibile porre vincoli sulla "correttezza" di un documento.

Il markup descrittivo (1)

Basato sul ruolo

- ◆ Di ogni elemento viene descritto il ruolo all'interno del testo, più che le regole per la sua visualizzazione:
- ◆ Ad esempio: “questo è un titolo, questo è un paragrafo, questo è il nome dell'autore, questa è una citazione.”

Indipendente dal sistema

- ◆ Poiché il markup descrittivo assegna ruoli (e non regole di visualizzazione) agli elementi del testo, questi sono intrinseci agli elementi stessi, e non alle funzionalità disponibili nel sistema di visualizzazione.
- ◆ Un sistema incapace di variare l'interlinea, o con un elenco limitato di font e dimensioni, può aver problemi ad interpretare un markup procedurale troppo ricco, ma la differenza tra (per esempio) un “titolo” o un “elenco” o un “paragrafo” non dipende dalla sofisticazione del sistema di visualizzazione.

Il markup descrittivo (2)

Basato su categorie

- ◆ I ruoli sono categorie. Ogni elemento è associato ad una categoria, e ne riflette tutte le caratteristiche automaticamente.

Contestuale

- ◆ Con il markup descrittivo è possibile definire delle regole che permettano o impediscano l'assegnazione di una categoria (ruolo) ad un elemento del testo a seconda del contesto.
- ◆ Ad esempio, si può richiedere che il titolo vada all'inizio del testo, o che una lista sia composta solo di elementi della lista, e non da paragrafi, ecc.
- ◆ Ancora, è possibile specificare *regole di correttezza* sui documenti, ad esempio che ad un'immagine segua **necessariamente** una didascalia, ecc.

Il markup descrittivo (3)

Il markup generico sfrutta il *late binding*: le scelte specifiche si fanno all'ultimo momento utile.

Nel nostro caso, la formattazione viene decisa nel momento in cui il documento viene visualizzato, o stampato, piuttosto che quando il documento viene codificato.

Per contro, il markup descrittivo richiede l'esistenza di due passaggi distinti, la marcatura e la formattazione, che usano due tecnologie diverse e vengono fatte in due momenti diversi.

Il markup descrittivo (4)

Ci sono vari vantaggi in questa tecnica:

- ◆ **Facilità nella creazione:** l'autore si concentra sul ruolo organizzativo delle singole parti di testo, piuttosto che sul loro aspetto stampato.
- ◆ **Indipendenza dalla formattazione:** riformattare un documento secondo nuove regole richiede semplicemente di ricodificare dei parametri esterni, non di modificare in alcuna maniera il documento
- ◆ **Flessibilità:** riusare un documento in un nuovo contesto è facile, perché non è necessario rimuovere la vecchia informazione per far posto alla nuova.
- ◆ **Visioni di documenti dinamicamente riconfigurabili:** è possibile evidenziare di volta in volta caratteristiche del documento diverse (caratteristica degli outline processor, per esempio)

Modi del markup (2)

Tradizionalmente, editor (sistemi di inserimento dati) e formatter (sistemi di creazione di output) erano separati. LaTeX ancora riflette questa distinzione, MS Word no.

Il markup viene utilizzato da un formatter per la creazione dell'output, ma l'editor può operare sul markup se ne conosce le caratteristiche. In questo caso il markup sarà:

- ◆ **Esposto**: il sistema mostra il markup e ne mostra gli effetti
- ◆ **Travestito**: il sistema mostra nel contenuto un simbolo che attiva il markup
- ◆ **Nascosto**: il sistema nasconde il markup e ne mostra solo gli effetti
- ◆ **Visualizzato**: il sistema non interpreta in markup e lo mostra insieme al contenuto.

Uso del markup (su computer)

- Possiamo sistemare, riorganizzare e formattare un testo per la stampa.
- Possiamo sistemare, riorganizzare e formattare un testo per la lettura a video.
- Possiamo sistemare, riorganizzare e formattare un testo per l'uso di handicappati fisici (ad esempio ciechi con l'aiuto di un lettore Braille)
- Possiamo sistemare, riorganizzare e formattare un testo per la lettura del testo ad alta voce in situazioni di impedimento temporaneo (ad esempio, mentre stiamo guidando).
- Possiamo permettere facilmente una ricerca sui contenuti del testo.
- Possiamo verificare la adeguatezza del testo rispetto a regole più o meno formali di strutturazione.

Conclusioni

Qui abbiamo parlato di

- ◆ Importanza del markup nel testo
- ◆ Rilevanza del markup descrittivo per applicare scopi multipli allo stesso testo
- ◆ Qualche distinzione tra tipi di markup

Riferimenti

- ◆ J. H. Coombs, A. H. Renear, S. J. DeRose, Markup Systems and the future of Scholarly Text Processing, *Communications of the ACM*, 30(11), November 1987.