

Rappresentazione dell'informazione

Notazione posizionale: valore cifra dipendente dalla posizione
(sistema decimale)

Numeri naturali

b (base)	2	8	10	16
insieme S	0, 1	0,1,2,3,4,5,6,7	0,1,2,3,4, 5,6,7,8,9	0,1,2,3,4,5,6,7,8,9, A,B,C,D,E,F
$c_i \in [0, b-1]$				

numero N (posizionale) = $c_{n-1} c_{n-2} \dots c_0$ con $c_i \in S$

Interpretazione in base decimale di un numero in base b

$$N_{10} = \sum_{i=0}^{n-1} c_i b^i, \text{ dove } n \equiv \text{numero cifre}$$

Es. $N_2 = 101_2 = 1 \cdot 2^2 + 0 \cdot 2^1 + 1 \cdot 2^0 = 5_{10}$

$$N_8 = 101_8 = 1 \cdot 8^2 + 0 \cdot 8^1 + 1 \cdot 8^0 = 65_{10}$$

$$N_{10} = 101_{10} = 1 \cdot 10^2 + 0 \cdot 10^1 + 1 \cdot 10^0 = 101_{10}$$

Conversione numero naturale da base 10 a base 2

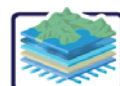
1. Numero iniziale diviso (divisione intera) ripetutamente per la base 2 fino a quando il risultato è 0;
2. i resti prodotti dalle divisioni intere presi in ordine inverso costituiscono la codifica cercata.

Osservazione: l'algoritmo converge sempre



POLITECNICO
DI MILANO

Politecnico di Milano – DEI –
Prof. Mauro Negri



SpatialDBgroup

Esempio: $N_{10} = 26$

base $\rightarrow 2$		resti	base 8	
26		0 - significativo	26	2
13		1	3	3
6		0	0	fine
3		1		
1		1+ significativo		
0		\leftarrow fine		

$$26_{10} = 11010_2 = 32_8 = 1A_{16}$$

Conversioni utili

$$\text{Binario} \longleftrightarrow \text{ottale} \quad 26_{10} = (0)11.010 = 3\ 2$$

$$\text{Binario} \longleftrightarrow \text{esadecimale} \quad 26_{10} = (000)1.1010 = 1\ A$$

Rappresentazione dei numeri naturali nel calcolatore

L'unità atomica di rappresentazione è la cifra binaria (base 2) chiamata bit (BInary digiT).

Un numero naturale binario è memorizzato in una parola di memoria composta da N bit (cifre).



Impossibile rappresentare tutti i numeri naturali: ∞ richiede ∞ bit

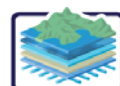


La rappresentazione finita riduce quindi l'intervallo rappresentabile



POLITECNICO
DI MILANO

Politecnico di Milano – DEI –
Prof. Mauro Negri



SpatialDBgroup

Intervallo di rappresentabilità

Dati N bit quanti e quali numeri naturali si possono rappresentare?

2^N combinazioni, ossia tutti i numeri nell'intervallo $[0, 2^N-1]$

N	2^N
10	1024 (1K)
16	65536(64K)
20	$\cong 1.000.000(M)$
32	$\cong 4$ miliardi (G)

Corrispondenza tra combinazione binaria e numero per N=16

0000 0000 0000 0000 = 0

0000 0000 0000 0001 = 1

0000 0000 0000 0010 = 2

...

0000 0000 0001 1010 = 26

1111 1111 1111 1111 = 65535

Osservazioni:

- l'intervallo di rappresentabilità è finito
- tutti i numeri all'interno dell'intervallo sono rappresentabili

Aritmetica binaria

Somma

00001 ←riporti

011001 + 25+

100001 = 33=

111010 58

Sottrazione

00100 ←prestiti

101001 - 41 -

100101 = 37 =

000100 4

Prodotto

010 x 2 x

101 = 5 =

010

000

010

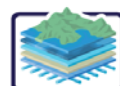
01010 10

Aumenta il num. di bit necessari ->



POLITECNICO
DI MILANO

Politecnico di Milano – DEI –
Prof. Mauro Negri



SpatialDBgroup

Osservazioni

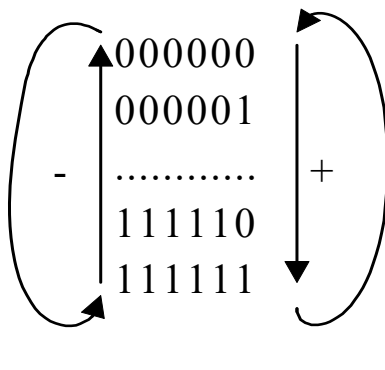
- possibile produzione di valori non rappresentabili (es. 65535+ 1)
- Ogni operazione produce sempre un risultato

```
1111 1111 1111 111
1111 1111 1111 1111 +
0000 0000 0000 0001 =
```

(1) 0000 0000 0000 0000

Ripporto (carry) -> overflow

Circolarità nel calcolo sommatore



Registro di stato della CPU_{xx} (SR)

	Z	N	C	V
--	---	---	---	---

Z=1 se addizione =0

N=1 se risultato addizione è negativo

C=1 se riporto sul bit + significativo in addizione (carry)

V=1 se overflow

Dimostrazione In limits.h unsigned short int (max=65535)

```
unsigned short int v=0;
```

```
.....
```

```
while (1)
```

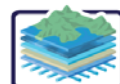
```
{v++; if (v==0){printf("\n v=0");break; }
```

```
else printf("\n%d", v); }
```



POLITECNICO
DI MILANO

Politecnico di Milano – DEI –
Prof. Mauro Negri



SpatialDBgroup

Exec

1
2
3
...
65534
65535
v=0

Es2. In limits.h short int (min=-32768, max=32767)

```
short int v=0;
while (1)
{v++; printf("\n%d", v);
  if(v<=0) { sleep (1);}
  if (v==0){printf("\n v=0");break; }
}
```

Exec

1	0000 0000 0000 0001
2	0000 0000 0000 0010
...	
32767	0111 1111 1111 1111
-32768	1000 0000 0000 0000
-32767	1000 0000 0000 0001
...	
-1	1111 1111 1111 1111

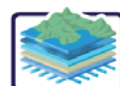
Errore assoluto è costante < 1

Errore relativo che diminuisce al crescere dei numeri.



POLITECNICO
DI MILANO

Politecnico di Milano – DEI –
Prof. Mauro Negri



SpatialDBgroup

Rappresentazione dei numeri interi

L'intervallo di rappresentabilità viene suddiviso tra numeri positivi e negativi

Esempi di intervalli f(hardware, linguaggio)

Ogni linguaggio, compilatore decide come gestire questi limiti.

Tipi C	Bit	Intervallo possibile (limits.h)
<i>short integer</i>	16	-32768 a 32767 <i>SHRT_MIN, SHRT_MAX</i>
<i>unsigned short integer</i>	16	0 a 65535 <i>USHRT_MAX</i>
<i>Integer</i>	32	-2147483648 a 2147483647 <i>INT_MIN, INT_MAX</i>
<i>unsigned integer</i>	32	0 a 4294967295 <i>UINT_MAX</i>
<i>long integer</i>	32	-2147483648 a 2147483647 <i>LONG_MIN, LONG_MAX</i>
<i>unsigned long integer</i>	32	0 a 4294967295 <i>ULONG_MAX</i>

Convenzione codifica in modulo e segno (N bits)

- bit + significativo: 1 (negativi), 0 (positivi)
- valori rappresentabili: $-(2^{N-1}) \longleftrightarrow (2^{N-1})$
- doppia rappresentazione dello 0

Esempio:

111(-3), 110(-2), 101(-1), 100(-0), 000(+0), 001(+1), 010(+2), 011(+3)

Codifica in complemento a 2

- si abbandona la regola del cambio base
- valori rappresentabili: $-(2^{N-1}) \longleftrightarrow (2^{N-1}-1)$
- primo bit mantiene il segno

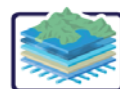
Esempio:

100(-4), 101(-3), 110(-2), 111(-1), 000(+0), 001(+1), 010(+2), 011(+3)



POLITECNICO
DI MILANO

Politecnico di Milano – DEI –
Prof. Mauro Negri



SpatialDBgroup

Come si calcola

Numeri positivi: codifica binaria del valore assoluto

Numeri negativi: $|N_2| + (N_2) = 0$

ES. -3

$$\begin{array}{r} 011 \text{ (3)} \\ 101 \text{ (-3)} \\ \hline (1)000 \end{array} \quad \rightarrow N_2 = 2^N - |N_2|$$
$$\begin{array}{r} 1000- \\ 011 \\ \hline 101 \end{array}$$

Regole immediate per numeri negativi

1. complemento a 1 di $+N_2 + 1$ ($011 \rightarrow 100 + 1 \rightarrow 101$)
2. ricopiare bit – significativi di N_2 sino a primo 1 incluso e poi complemento a 1 dei rimanenti.

Osservazioni:

- unica rappresentazione dello 0 (complemento a 2)
- sottrazione realizzata come somma

Torniamo all'overflow di $A + B$

Come vengono caricati i bit C e V

$A > 0, B > 0$

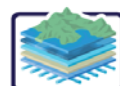
- $C = 0$
- se risultato ha primo bit a 0 \Rightarrow no overflow altrimenti $V = 1$

$$\begin{array}{r} 011 + 3 \\ 011 = 3 \\ \hline 110 - 2 \end{array}$$



POLITECNICO
DI MILANO

Politecnico di Milano – DEI –
Prof. Mauro Negri



SpatialDBgroup

$$A > 0, B < 0 \text{ (o viceversa)} \quad A + (2^N - |B|) = 2^N - (|B| - A)$$

- $V=0$

- $|B| > A \rightarrow C=0$

$$010 + 2$$

$$101 = -3$$

$$111 \quad -1$$

- $|B| \leq A \rightarrow C=1$ corretto dopo eliminazione riporto

$$011 + (3)$$

$$011 + (3)$$

$$101 = (-3)$$

$$110 = (-2)$$

$$(1)000$$

$$(1)001$$

$$A < 0, B < 0 \quad (2^N - |A|) + (2^N - |B|) = 2^N + 2^N - (|B| + |A|)$$

- $C=1$ riporto è strutturale e quindi non significativo

- se risultato ha primo bit a 1 \Rightarrow no overflow altrimenti $V=1$

$$V=0 \quad 111 + -1$$

$$101 = -3$$

$$(1)100 \quad -4$$

$$V=1 \quad 101 + (-3)$$

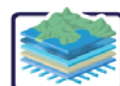
$$101 = (-3)$$

$$010 \rightarrow +2$$



POLITECNICO
DI MILANO

Politecnico di Milano – DEI –
Prof. Mauro Negri



SpatialDBgroup

Rappresentazione dei caratteri

- lettere maiuscole/minuscole A a .. Z z
- spazio
- cifre 0 9
- segni di interpunzione , : ; .
- simboli ! " # % @) < =
- caratteri di controllo per gestire la visualizzazione, la stampa, la trasmissione dei caratteri (*inizio riga, salto di riga, salto pagina*)

Codifica ASCII (American Standard Code for Information Interchange) basata su 8 bit

- da 0 a 32 caratteri speciali non stampabili
- 65 A, 66 B, 67 C, 97 a, 98 b ordinamento per concetto di precede/segue e distanza tra due lettere e tra la stessa lettera minuscola e maiuscola (costante)
- lettere senza accento (alfabeto inglese)
- 128 configurazioni perse

codifiche nazionali nell'intervallo 128-255 \Rightarrow portabilità

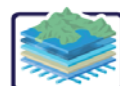
La dualità carattere e intero

```
char b='z';    printf("%c %d",b,b);                    z    122
```



POLITECNICO
DI MILANO

Politecnico di Milano – DEI –
Prof. Mauro Negri



SpatialDBgroup

ASCII TABLE

Decimal	Hex	Char	Decimal	Hex	Char	Decimal	Hex	Char	Decimal	Hex	Char
0	0	[NULL]	32	20	[SPACE]	64	40	@	96	60	,
1	1	[START OF HEADING]	33	21	!	65	41	A	97	61	a
2	2	[START OF TEXT]	34	22	"	66	42	B	98	62	b
3	3	[END OF TEXT]	35	23	#	67	43	C	99	63	c
4	4	[END OF TRANSMISSION]	36	24	\$	68	44	D	100	64	d
5	5	[ENQUIRY]	37	25	%	69	45	E	101	65	e
6	6	[ACKNOWLEDGE]	38	26	&	70	46	F	102	66	f
7	7	[BELL]	39	27	'	71	47	G	103	67	g
8	8	[BACKSPACE]	40	28	(72	48	H	104	68	h
9	9	[HORIZONTAL TAB]	41	29)	73	49	I	105	69	i
10	A	[LINE FEED]	42	2A	*	74	4A	J	106	6A	j
11	B	[VERTICAL TAB]	43	2B	+	75	4B	K	107	6B	k
12	C	[FORM FEED]	44	2C	,	76	4C	L	108	6C	l
13	D	[CARRIAGE RETURN]	45	2D	.	77	4D	M	109	6D	m
14	E	[SHIFT OUT]	46	2E	:	78	4E	N	110	6E	n
15	F	[SHIFT IN]	47	2F	/	79	4F	O	111	6F	o
16	10	[DATA LINK ESCAPE]	48	30	0	80	50	P	112	70	p
17	11	[DEVICE CONTROL 1]	49	31	1	81	51	Q	113	71	q
18	12	[DEVICE CONTROL 2]	50	32	2	82	52	R	114	72	r
19	13	[DEVICE CONTROL 3]	51	33	3	83	53	S	115	73	s
20	14	[DEVICE CONTROL 4]	52	34	4	84	54	T	116	74	t
21	15	[NEGATIVE ACKNOWLEDGE]	53	35	5	85	55	U	117	75	u
22	16	[SYNCHRONOUS IDLE]	54	36	6	86	56	V	118	76	v
23	17	[ENG OF TRANS. BLOCK]	55	37	7	87	57	W	119	77	w
24	18	[CANCEL]	56	38	8	88	58	X	120	78	x
25	19	[END OF MEDIUM]	57	39	9	89	59	Y	121	79	y
26	1A	[SUBSTITUTE]	58	3A	:	90	5A	Z	122	7A	z
27	1B	[ESCAPE]	59	3B	;	91	5B	[123	7B	{
28	1C	[FILE SEPARATOR]	60	3C	<	92	5C	\	124	7C	
29	1D	[GROUP SEPARATOR]	61	3D	=	93	5D]	125	7D	}
30	1E	[RECORD SEPARATOR]	62	3E	>	94	5E	^	126	7E	~
31	1F	[UNIT SEPARATOR]	63	3F	?	95	5F	_	127	7F	[DEL]

Da uno standard per l'inglese e per la stampa ad uno per l'interoperabilità estesa ad altri alfabeti

ISO 8859 standard per usare ottavo bit, ma 128 carattere sono pochi se si esce dall'inglese (8859-1 sino a 8859-16 di cui 1 abbandonata)

ISO 8859-1 (latin 1) Europa Occidentale

- Lingue - anche parzialmente coperte: [danese](#), [finlandese](#), [francese](#), [gaelico scozzese](#), [inglese](#), [irlandese](#), [islandese](#), [italiano](#), [norvegese](#), [olandese](#), [portoghese](#), [romancio](#), [spagnolo](#), [svedese](#) e [tedesco](#), [albanese](#),
- € in revisione ISO 8859-15 versione riveduta. Il corrispondente repertorio di caratteri approvato dall'IANA ISO-8859-1
- è la codifica predefinita per documenti [HTML](#) e per documenti trasmessi tramite messaggi [MIME](#), come le risposte [HTTP](#) quando il tipo documento è "text" (come in "text/html").

Unicode (1991)

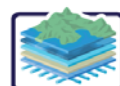
Obiettivi

- consistenza con ASCII
- creare l'universal character set
- raggruppiamo alfabeti in european (African,...) scripts
- European scripts sono:
Armenian, Cyrillic, Greek, Latin-1 (Basic latin o ASCII) Latin 1 supplemented,.....



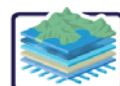
POLITECNICO
DI MILANO

Politecnico di Milano – DEI –
Prof. Mauro Negri

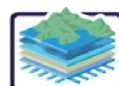


SpatialDBgroup

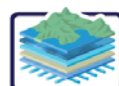
	000	001	002	003	004	005	006	007
0	NUL 0000	DLE 0010	SP 0020	0 0030	@ 0040	P 0050	` 0060	p 0070
1	SOH 0001	DC1 0011	! 0021	1 0031	A 0041	Q 0051	a 0061	q 0071
2	STX 0002	DC2 0012	" 0022	2 0032	B 0042	R 0052	b 0062	r 0072
3	ETX 0003	DC3 0013	# 0023	3 0033	C 0043	S 0053	c 0063	s 0073
4	EOT 0004	DC4 0014	\$ 0024	4 0034	D 0044	T 0054	d 0064	t 0074
5	ENQ 0005	NAK 0015	% 0025	5 0035	E 0045	U 0055	e 0065	u 0075
6	ACK 0006	SYN 0016	& 0026	6 0036	F 0046	V 0056	f 0066	v 0076
7	BEL 0007	ETB 0017	' 0027	7 0037	G 0047	W 0057	g 0067	w 0077
8	BS 0008	CAN 0018	(0028	8 0038	H 0048	X 0058	h 0068	x 0078
9	HT 0009	EM 0019) 0029	9 0039	I 0049	Y 0059	i 0069	y 0079
A	LF 000A	SUB 001A	* 002A	: 003A	J 004A	Z 005A	j 006A	z 007A
B	VT 000B	ESC 001B	+ 002B	; 003B	K 004B	[005B	k 006B	{ 007B
C	FF 000C	FS 001C	, 002C	< 003C	L 004C	\ 005C	l 006C	 007C
D	CR 000D	GS 001D	- 002D	= 003D	M 004D] 005D	m 006D	} 007D
E	SO 000E	RS 001E	. 002E	> 003E	N 004E	^ 005E	n 006E	~ 007E
F	SI 000F	US 001F	/ 002F	? 003F	O 004F	_ 005F	o 006F	DEL 007F



	008	009	00A	00B	00C	00D	00E	00F
0	XXX 0080	DCS 0090	NB SP 00A0	° 00B0	À 00C0	Đ 00D0	à 00E0	đ 00F0
1	XXX 0081	PU1 0091	¡ 00A1	± 00B1	Á 00C1	Ñ 00D1	á 00E1	ñ 00F1
2	BPH 0082	PU2 0092	¢ 00A2	² 00B2	Â 00C2	Ò 00D2	â 00E2	ò 00F2
3	NBH 0083	STS 0093	£ 00A3	³ 00B3	Ã 00C3	Ó 00D3	ã 00E3	ó 00F3
4	IND 0084	CCH 0094	¤ 00A4	´ 00B4	Ä 00C4	Ô 00D4	ä 00E4	ô 00F4
5	NEL 0085	MW 0095	¥ 00A5	µ 00B5	Å 00C5	Õ 00D5	å 00E5	õ 00F5
6	SSA 0086	SPA 0096	¦ 00A6	¶ 00B6	Æ 00C6	Ö 00D6	æ 00E6	ö 00F6
7	ESA 0087	EPA 0097	§ 00A7	· 00B7	Ç 00C7	× 00D7	ç 00E7	÷ 00F7
8	HTS 0088	SOS 0098	¨ 00A8	¸ 00B8	È 00C8	Ø 00D8	è 00E8	ø 00F8
9	HTJ 0089	XXX 0099	© 00A9	¹ 00B9	É 00C9	Ù 00D9	é 00E9	ù 00F9
A	VTS 008A	SCI 009A	 00AA	º 00BA	Ê 00CA	Ú 00DA	ê 00EA	ú 00FA
B	PLD 008B	CSI 009B	« 00AB	» 00BB	Ë 00CB	Û 00DB	ë 00EB	û 00FB
C	PLU 008C	ST 009C	¬ 00AC	¼ 00BC	Ì 00CC	Ü 00DC	ì 00EC	ü 00FC
D	RI 008D	OSC 009D	SHY 00AD	½ 00BD	Í 00CD	Ý 00DD	í 00ED	ý 00FD
E	SS2 008E	PM 009E	® 00AE	¾ 00BE	Î 00CE	Þ 00DE	î 00EE	þ 00FE
F	SS3 008F	APC 009F	¯ 00AF	¿ 00BF	Ï 00CF	ß 00DF	ï 00EF	ÿ 00FF



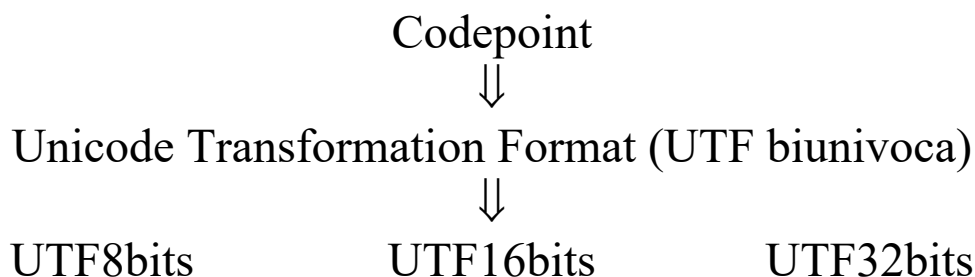
	010	011	012	013	014	015	016	017
0	Ā 0100	Ð 0110	Ġ 0120	İ 0130	Ĭ 0140	Ō 0150	Š 0160	Ū 0170
1	ā 0101	đ 0111	ġ 0121	ı 0131	ĭ 0141	ō 0151	š 0161	ū 0171
2	Ă 0102	Ē 0112	Ģ 0122	Ĳ 0132	ł 0142	Œ 0152	Ţ 0162	Ț 0172
3	ă 0103	ē 0113	ģ 0123	ij 0133	Ń 0143	œ 0153	ţ 0163	ț 0173
4	Ą 0104	Ě 0114	Ĥ 0124	Ĵ 0134	ń 0144	Ŕ 0154	Ť 0164	Ŵ 0174
5	ą 0105	ě 0115	ĥ 0125	ĵ 0135	Ň 0145	ř 0155	ť 0165	ŵ 0175
6	Ć 0106	Ė 0116	Ħ 0126	Ɔ 0136	ņ 0146	Ŗ 0156	Ŧ 0166	Ŷ 0176
7	ć 0107	ė 0117	ħ 0127	ķ 0137	Ņ 0147	ŗ 0157	ŧ 0167	ŷ 0177
8	Ĉ 0108	Ɛ 0118	Ĩ 0128	κ 0138	ň 0148	Ř 0158	Ũ 0168	Ÿ 0178
9	ĉ 0109	ɛ 0119	ĩ 0129	Ł 0139	'n 0149	ř 0159	ũ 0169	Ž 0179
A	Ċ 010A	Ė 011A	Ī 012A	Í 013A	Ņ 014A	Ś 015A	Ū 016A	ž 017A
B	ċ 010B	ė 011B	ī 012B	Ĳ 013B	ņ 014B	ś 015B	ū 016B	Ž 017B
C	Č 010C	Ĝ 011C	Ĭ 012C	ı 013C	Ō 014C	Ŝ 015C	Ŭ 016C	ž 017C
D	č 010D	ĝ 011D	ĭ 012D	Ĳ 013D	ō 014D	ŝ 015D	ŭ 016D	Ž 017D
E	Ď 010E	Ģ 011E	Ĳ 012E	Ĳ 013E	Ō 014E	Ŝ 015E	Ŭ 016E	ž 017E
F	ď 010F	ģ 011F	ĳ 012F	Ĳ 013F	ǫ 014F	ŝ 015F	ŭ 016F	f 017F



Rappresentazione logica del carattere

- Ad ogni carattere è associato un codepoint
- Un codepoint = <uniqueNumberValue, name>
where
numbervalue espresso in esadecimale utilizzando prima 16 e ora 21 bits
Es., U+0041 -65dec –A
Name descrizione
Es., Latin Capital Letter A

Implementazione fisica del codepoint



UTF16 bit distribution

Scalar Value	UTF-16
XXXXXXXXXXXXXXXXXX	XXXXXXXXXXXXXXXXXX
000uuuuuXXXXXXXXXXXXXXXXXX	110110wwwXXXXXX 110111XXXXXXXXXX

Note: www = uuuuu - 1

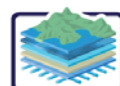
UTF8 bit distribution

Scalar Value	First Byte	Second Byte	Third Byte	Fourth Byte
00000000 0xxxxxxx	0xxxxxxx			
00000yyy yyxxxxxx	110yyyyy	10xxxxxx		
zzzyyyy yyxxxxxx	1110zzzz	10yyyyyy	10xxxxxx	
000uuuuu zzzzyyyy yyxxxxxx	11110uuu	10uuzzzz	10yyyyyy	10xxxxxx



POLITECNICO
DI MILANO

Politecnico di Milano – DEI –
Prof. Mauro Negri



SpatialDBgroup

UTF8 Examples

codepoint	Carattere	Codeunits
U+004D	M	4D
U+0431 (1)	Ѓ	D0 B1
U+4E8B (2)	事	E4 BA 8B

(1) 0430 = 0000 0**100** **00**11 0001

Schema 0000 0yyy yyxxxxxx

↓

110yyyyy 10xxxxxx

↓

110**1.0000** 1011.0001

↓

D 0 B 1

(2) 4E8B = 0**100** **1110** **10**00 1011

Schema zzzzyyyy yyxxxxxx

↓

1110zzzz 10yyyyyy 10xxxxxx

↓

1110.0**100** .10**11.1010** 1000.1011

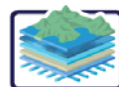
↓

E 4 B A 8 B



POLITECNICO
DI MILANO

Politecnico di Milano – DEI –
Prof. Mauro Negri



SpatialDBgroup

Rappresentazioni di immagini come bitmap

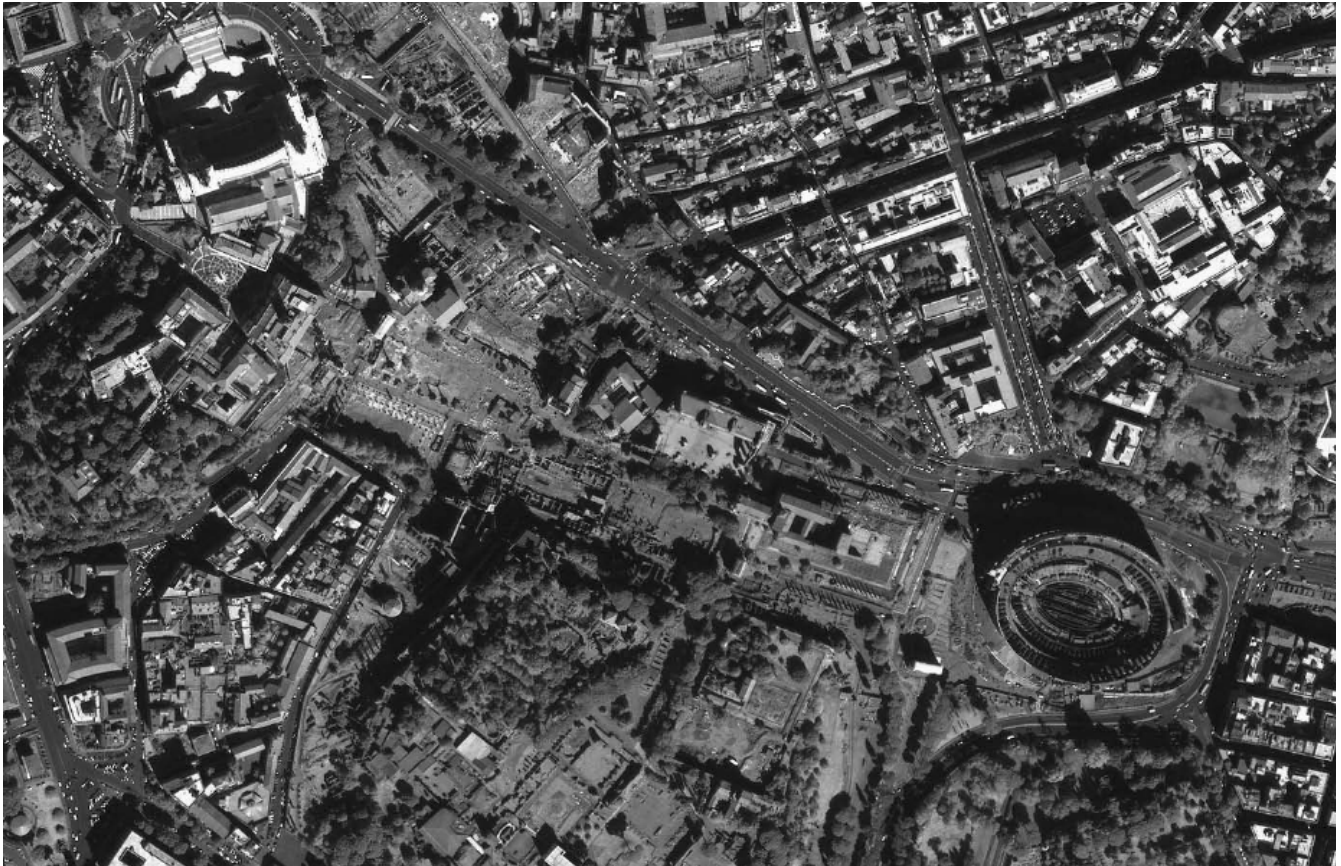


Immagine → digitalizzazione (cm x cm → punti x punti)
↑
risoluzione dpi(dot per inch - es. 1200 per fotogrammetria)

Numero bit x punto (pixel):

1. bianco/nero 1 bit x pixel
2. scala grigi (256 toni) 8 bit x pixel
3. colori a video: sistema RGB (256 toni per colore) 8 bit x 3 colori (24 bit)

Dimensioni: area(pixel * pixel)*bit x punto.

Es

schermo colori: $(1400 * 1050) * 3 * 8 = 35.280.000$ bits ~ 4,4Mbytes

Distanza di distinguibilità

2,56 cm (pollice) * 1/dpi (ad es., 1200) $\cong 21\mu\text{m}$

Fotogrammetria: risoluzione al suolo

scala	scala media fotogramma	risoluzione al suolo
1000	4500	$21\mu\text{m} * 4500 = 94500\mu\text{m} \cong 9,5\text{cm}$
2000	7000	$21\mu\text{m} * 7000 = 147000\mu\text{m} \cong 14,7\text{cm}$
5000	13000	$21\mu\text{m} * 13000 = 273000\mu\text{m} \cong 27,3\text{cm}$
10000	22000	$21\mu\text{m} * 22000 = 462000\mu\text{m} \cong 46,2\text{cm}$

Tecniche di compressione

senza
TIFF

con perdita di informazione
JPEG



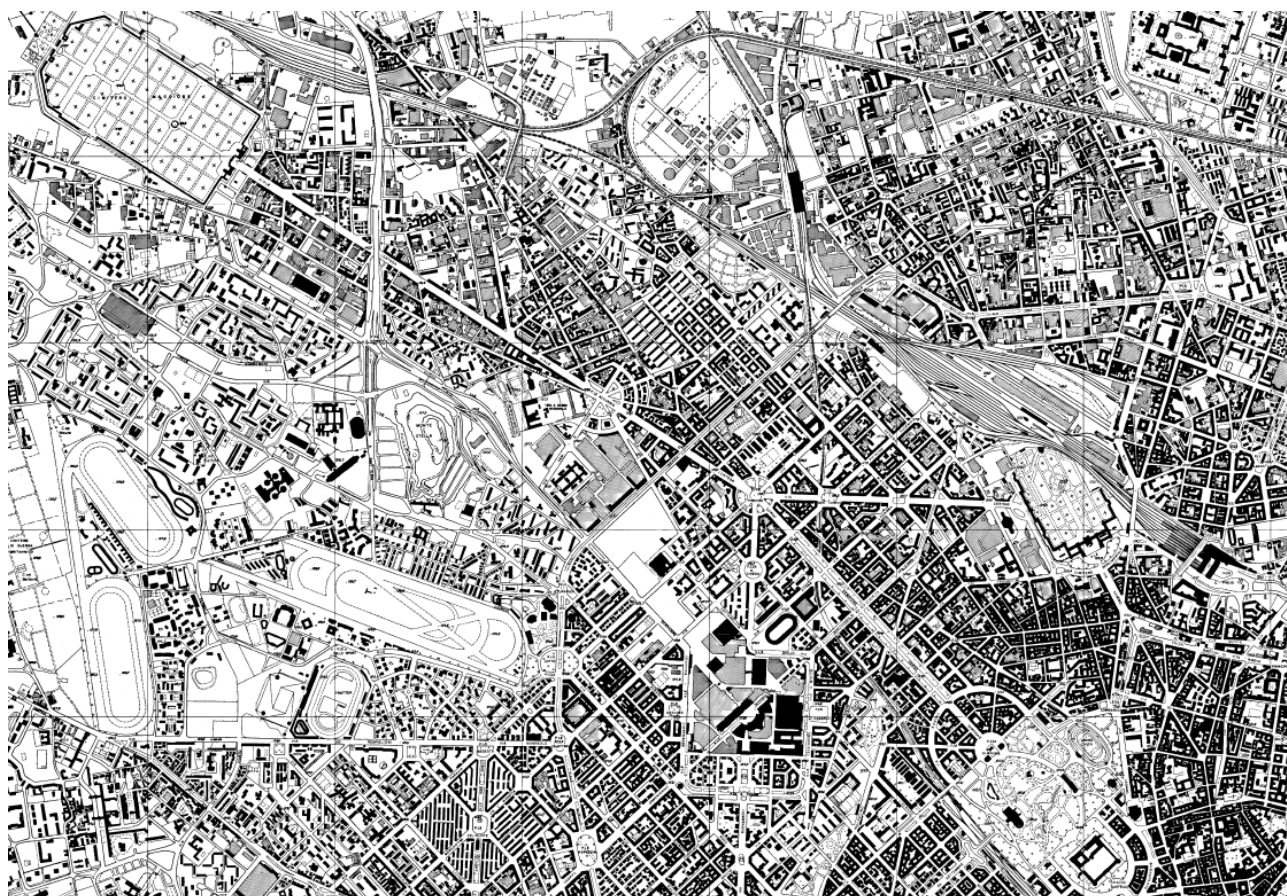
POLITECNICO
DI MILANO

Politecnico di Milano – DEI –
Prof. Mauro Negri

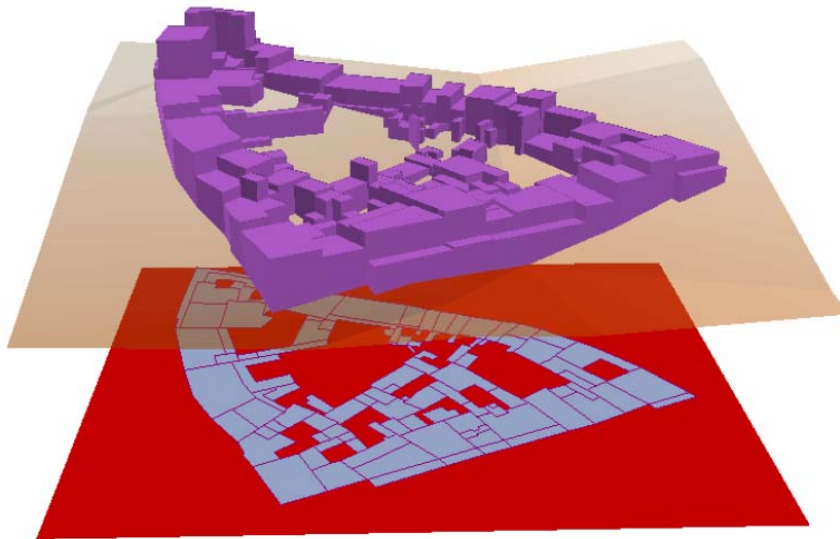


SpatialDBgroup

Dalle immagini alla sintesi vettoriale (cartografia)

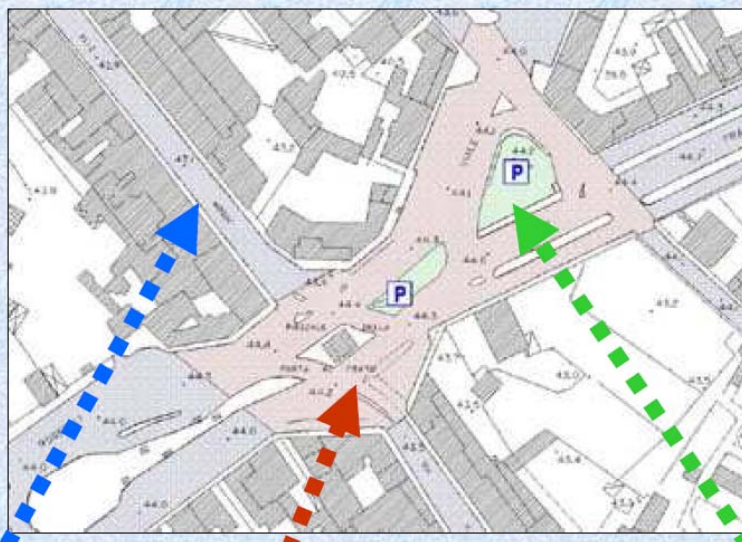


EDIFICIO e VOLUMETRIE



modellazione 3D

STRADE - AREA CIRCOLAZIONE VEICOLARE



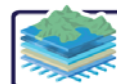
Tronco di
Carreggiata

Area Traffico
Strutturato
(es. INCROCIO)

Area a Traffico non
Strutturato
(es. PARCHEGGIO)



POLITECNICO
DI MILANO



SpatialDBgroup

Rappresentazione dei video

Video: sequenza di immagini

Sistema PAL: 720x576 pixel - 25 frame/sec.

$$(720 \cdot 576 \cdot 3 \cdot 8) \cdot 25 = (1,244 \text{MB}) \cdot 25 = \sim 31 \text{MB/sec}$$

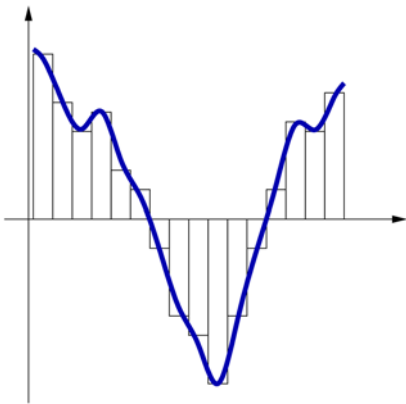
Film 133 minuti = 7980'' = $\sim 247 \text{GB}$ (29 DVD SSDL-8,5GB)

Tecniche di compressione variabile MPEG-2

- variazione tra fotogramma i -esimo e $(i-1)$ -esimo
- velocità media 3,5Mb/sec

Film 133 minuti = 7980'' $\cdot 3,5 = \sim 28 \text{Gb} = \sim 3,5 \text{GB}$ (1,2GB restanti per sottotitoli, lingue) \Rightarrow 1 DVD SSSL-4,7GB)

Rappresentazione del suono



Frequenza campionamento = $2 \cdot 22.000 \text{Hz} = \sim 44.100 \text{campioni/sec}$

Numero bit per campione per canale: 16

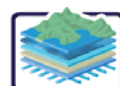
Dimensione = $44.100 \cdot 16 \cdot 2 = 1,4 \text{Mb/sec} = 172 \text{KB/sec}$

Voce (freq. Camp. $2 \cdot 4.000 \text{Hz}$)



POLITECNICO
DI MILANO

Politecnico di Milano – DEI –
Prof. Mauro Negri



SpatialDBgroup