Blockchain Decalogue

What Would Wei Wu Say on Blockchain

区块链津逮：十堂公开课与一首无止境的技术诗

AlphaWallet × 网易云课堂

Oct 18 ～ Nov 29, Weekly on every Thursday, 18:30 PM

PayPal Lab, Singapore

Ethereum Community Fund · DUO · BLOCK ASIA · BitTemple · COINVOICE 区块链创新者社区 · CHAINTECH · Bplus · ETHFANS · 仙女座科技 Andoromeda · Vietnam Blockchain Developers.

Celer · NERVOS · DBA · Taipei Ethereum · VeloxChain · GAM · 橙皮书 orangepaper · 火星财经 · 链闻 CHAIN NEWS · MIXMARVEL · bitfwd

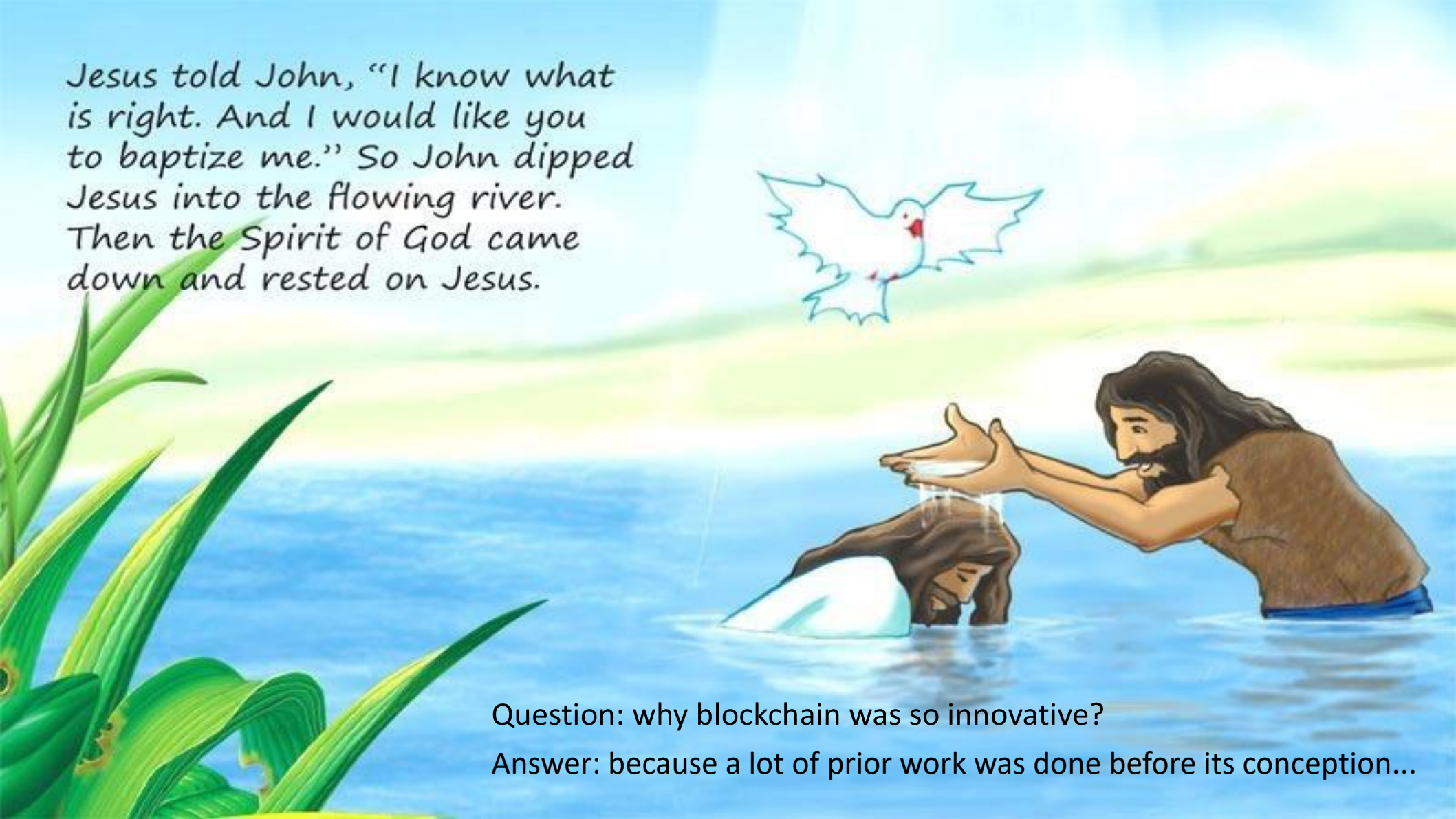
Talk 1: Elliptic curve and what was achieved before blockchain

Ask yourself, which of the following are unique to blockchain?

a) integrity (data can be tested against changes).

b) authenticity (only the keyholder could have done it).

c) Non-repudiation (undeniable proof);



Question: why blockchain was so innovative?

Answer: because a lot of prior work was done before its conception...

Properties of a digital signature scheme

a) integrity (data can be tested against changes).

b) authenticity (only the keyholder could've done it).

c) Non-repudiation (the keyholder couldn't deny having done it);

online. You are chatting with Alice.You are chatting with Alice.

Alice: Hi how can I help you today.

Bob: I wish to change my number to a new SIM card. The card number is 897283728374827338 1.897283728374827338 1.

Alice: Great I can certainly help you with that. Can you tell me your name, date of birth and your support code?date of birth and your support code?

Bob: Sure. 08-AUG-1988, Bob Boon, my secret code is Satoshi Nakamoto.Bob: Sure. 08-AUG-1988, Bob Boon, my secret code is Satoshi Nakamoto.

Alice: Perfect. Thank you Mr Boon. Your number is transferred to the new Alice: Perfect. Thank you Mr Boon. Your number is transferred to the new SIM card. What else can I do for you today?SIM card. What else can I do for you today?

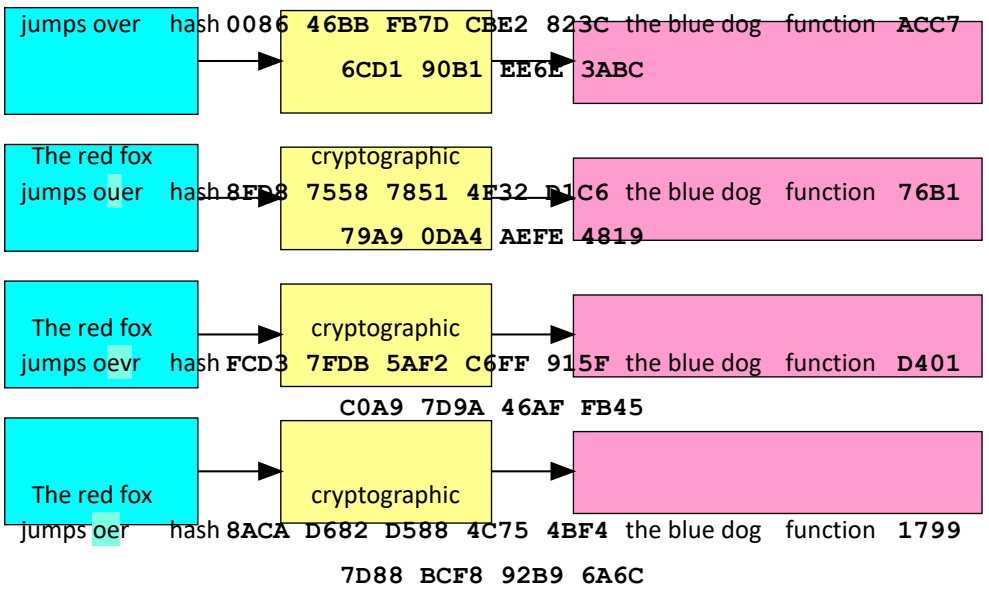
Bob: That's all good thanks!Bob: That's all good thanks!

Alice: Have a nice day! Please rate my service from 1 to 5.Alice: Have a nice day! Please rate my service from 1 to 5.

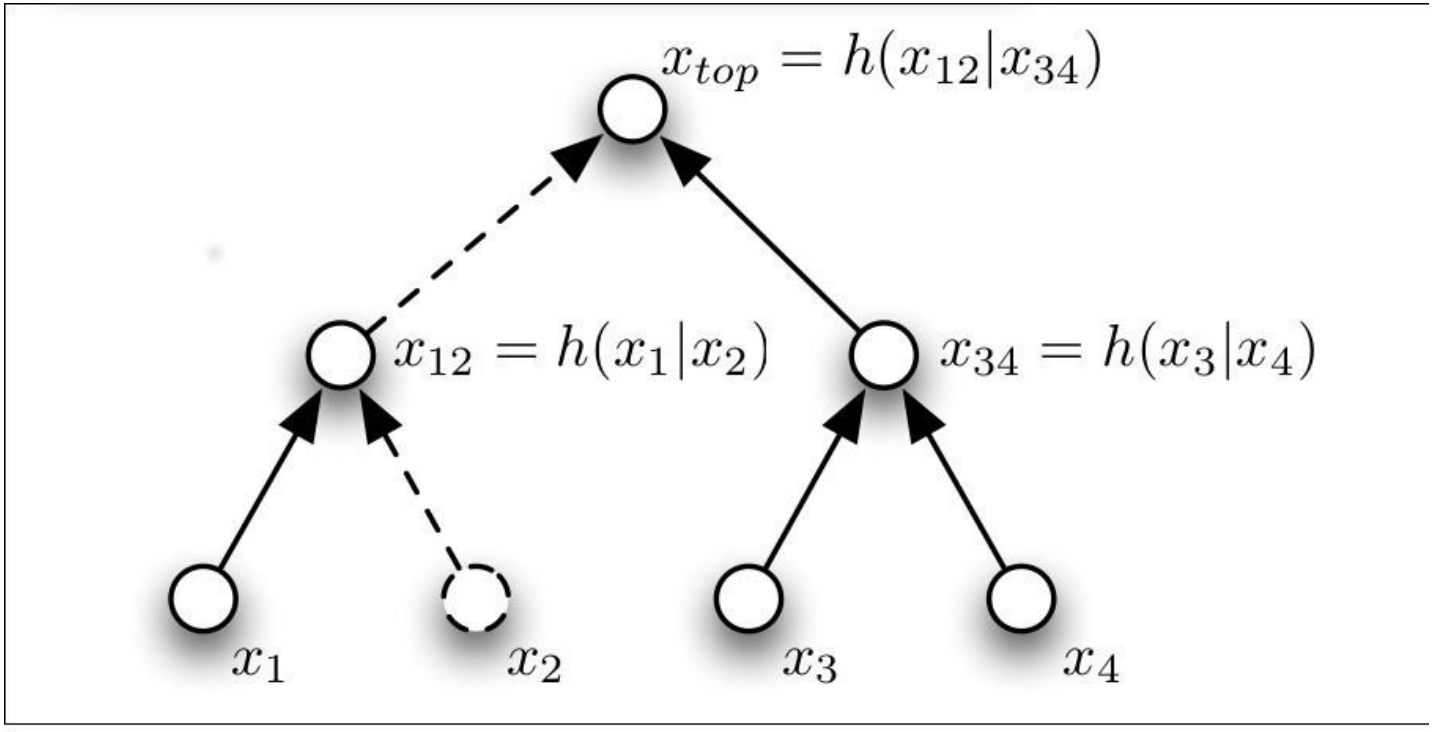
[Bob left the conversation[Bob left the conversation]]

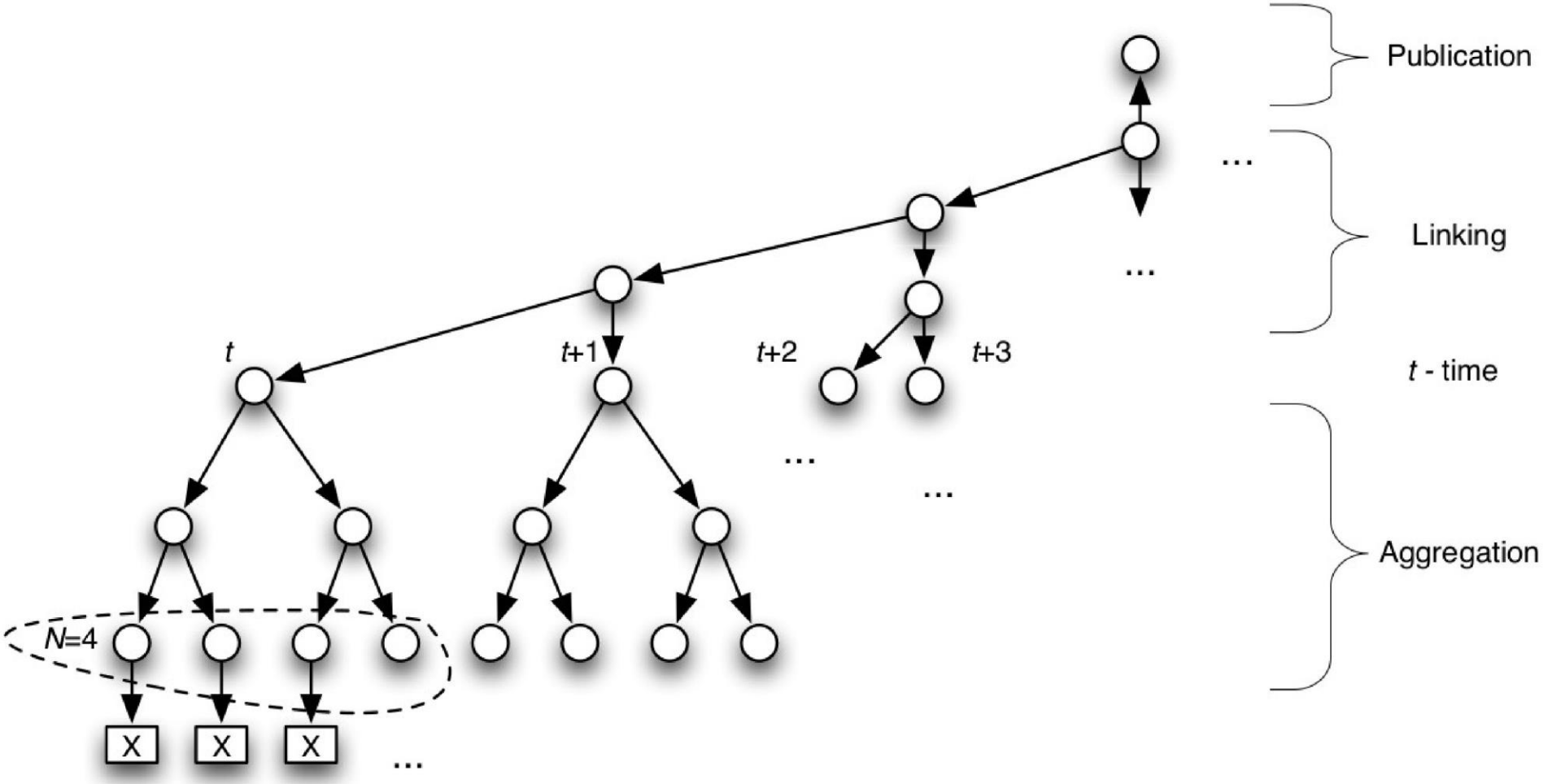
Data integrity 1: Hash Function

Input		Digest
Fox	cryptographic hash function	DFCD 3454 BBEA 788A 751A 696C 24D9 7009 CA99 2D17
The red fox	cryptographic	



https://en.wikipedia.org/wiki/File:Cryptographic_Hash_Function.svg


$x_{top} = h(x_{12}|x_{34})$
$x_{12} = h(x_1|x_2)$
$x_{34} = h(x_3|x_4)$
x_1
x_2
x_3
x_4


Publication
Linking
t - time
Aggregation
t
t+1
t+2
t+3
N=4
X
X
X
...

A ladder approach to one way function and hash function

Jacob's Dream, 1705, James Thornhill

38742

0489=

x^x

We’re here

Message to be hashed

```
$ echo -n "I love Singapore." | xxd -p
```

```
49206C6F766520536696E6761706F72652E
$ dc -e '16 i 9 o 49206C6F766520536696E6761706F72652E p'
20631354867572172766842757540751687131786 26
$ dc -e '16 i 4 o 49206C6F766520536696E6761706F72652E p' | tr '0123'
'\+\-\*\/'
-+*-+*++-*/+-*//-/-*-*--+*++--+/-***--*/*-*-/-*+--/++-*//-
/+*-*--+*/* $ dc -e '4 2-6+6*7-6+2+5*6+6+6-6*7/6+7-
6*2/20/54-72/66-57*51-31*6/p'
-19700$
```

Message to be hashed

```
$ echo -n "I love S'pore." | xxd -p
4920686F74652053E28099706F72652E
$ dc -e '16i 9o 4920686F74652053E28099706F72652E p'
58235668606420026063068625076467144745 42
$ dc -e '16i 4o 4920686F74652053E28099706F72652E p' | tr '0123'
'\+\-\*\/'
-+*-+*++-**+-*//-/-+-*--+*++--+//*+**+++*-*--/++-*//-/+*-*--+*/*
$ dc -e '4 2-6+6*7-6+2   -E+8-9*7*6-7*6/2/   -68/20-63+25-
```

```
$ bc <<< 'g=5; q=277; for (i=1; i<q; i++) {g^i%q}' | uniq -c | column -c 250 | sed
's/\t */\t/g' | expand -t6
```

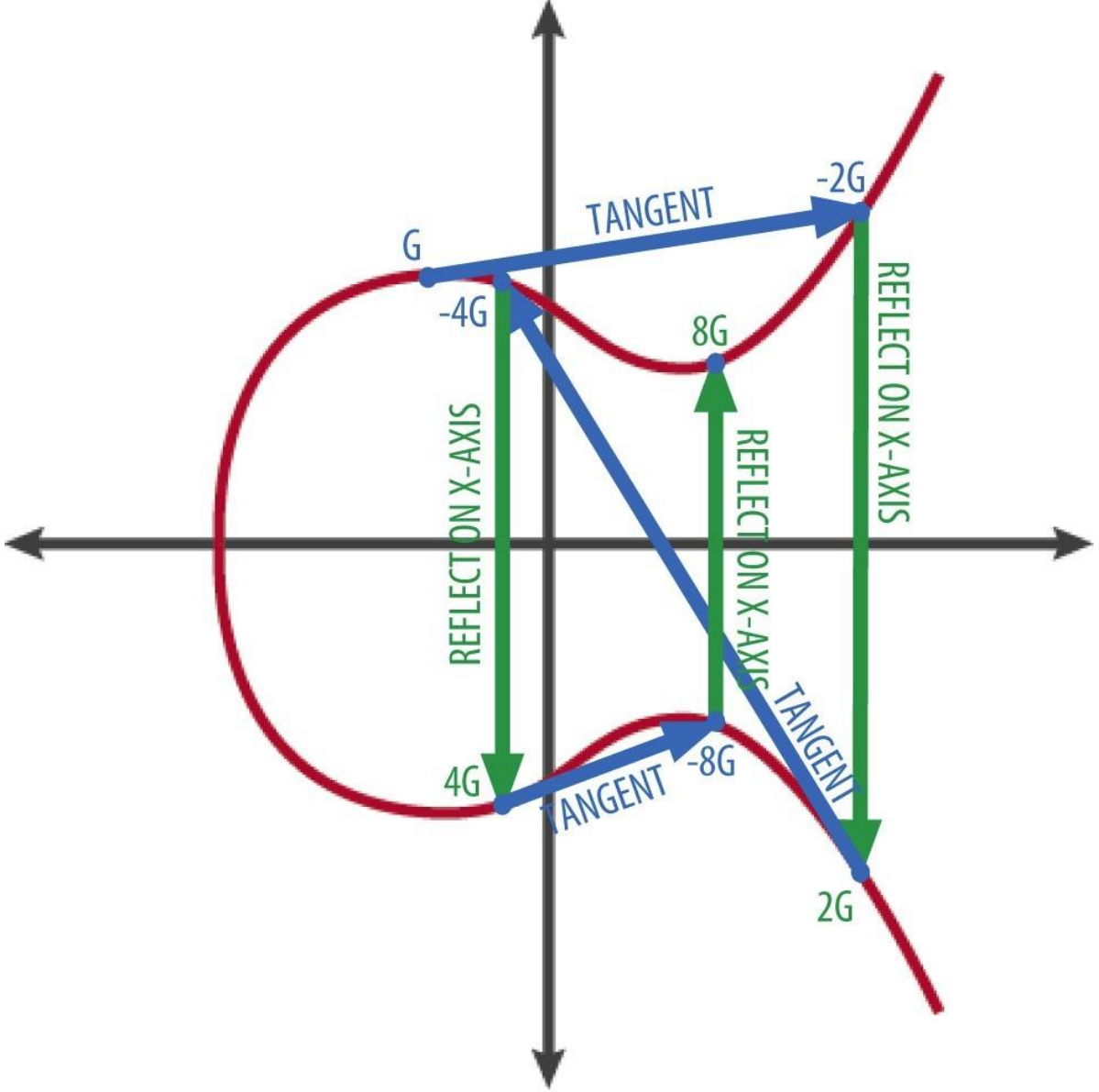
```
      1 5   1 100 1 61  1 112 1 24  1 203 1 182 1 39  1 226 1 88  1 98  1 21  1 143 1
90  1 138
      1 25  1 223 1 28  1 6   1 120 1 184 1 79  1 195 1 22  1 163 1 213 1 105 1 161 1
173 1 136
      1 125 1 7   1 140 1 30  1 46  1 89  1 118 1 144 1 110 1 261 1 234 1 248 1 251 1
34  1 126
      1 71  1 35  1 146 1 150 1 230 1 168 1 36  1 166 1 273 1 197 1 62  1 132 1 147 1
170 1 76
      1 78  1 175 1 176 1 196 1 42  1 9   1 180 1 276 1 257 1 154 1 33  1 106 1 181 1
19  1 103
      1 113 1 44  1 49  1 149 1 210 1 45  1 69  1 272 1 177 1 216 1 165 1 253 1 74  1
95  1 238
      1 11  1 220 1 245 1 191 1 219 1 225 1 68  1 252 1 54  1 249 1 271 1 157 1 93  1
198 1 82
      1 55  1 269 1 117 1 124 1 264 1 17  1 63  1 152 1 270 1 137 1 247 1 231 1 188 1
159 1 133
      1 275 1 237 1 31  1 66  1 212 1 85  1 38  1 206 1 242 1 131 1 127 1 47  1 109 1
241 1 111
      1 267 1 77  1 155 1 53  1 229 1 148 1 190 1 199 1 102 1 101 1 81  1 235 1 268 1
97  1 1
      1 227 1 108 1 221 1 265 1 37  1 186 1 119 1 164 1 233 1 228 1 128 1 67  1 232 1
208
      1 27  1 263 1 274 1 217 1 185 1 99  1 41  1 266 1 57  1 32  1 86  1 58  1 52  1
209
```

```
      1 135 1 207 1 262 1 254 1 94  1 218 1 205 1 222 1 8   1 160 1 153 1 13  1 260 1
214
      1 121 1 204 1 202 1 162 1 193 1 259 1 194 1 2   1 40  1 246 1 211 1 65  1 192 1
239
      1 51  1 189 1 179 1 256 1 134 1 187 1 139 1 10  1 200 1 122 1 224 1 48  1 129 1
87
      1 255 1 114 1 64  1 172 1 116 1 104 1 141 1 50  1 169 1 56  1 12  1 240 1 91  1
158
      1 167 1 16  1 43  1 29  1 26  1 243 1 151 1 250 1 14  1 3   1 60  1 92  1 178 1
236
      1 4   1 80  1 215 1 145 1 130 1 107 1 201 1 142 1 70  1 15  1 23  1 183 1 59  1
72
      1 20  1 123 1 244 1 171 1 96  1 258 1 174 1 156 1 73  1 75  1 115 1 84  1 18  1
```

```
$ bc <<< 'g=5; q=277; for (i=1; i<q; i++) {g^i%q}' | sort -n| uniq -c | column -c 250
| sed 's/\t */\ t/g' | expand -t6
      1 1   1 20  1 39  1 58  1 77  1 96  1 115 1 134 1 153 1 172 1 191 1 210 1 229 1
248 1 267
      1 2   1 21  1 40  1 59  1 78  1 97  1 116 1 135 1 154 1 173 1 192 1 211 1 230 1
249 1 268
      1 3   1 22  1 41  1 60  1 79  1 98  1 117 1 136 1 155 1 174 1 193 1 212 1 231 1
250 1 269
      1 4   1 23  1 42  1 61  1 80  1 99  1 118 1 137 1 156 1 175 1 194 1 213 1 232 1
251 1 270
```

```
      1 5   1 24  1 43  1 62  1 81  1 100 1 119 1 138 1 157 1 176 1 195 1 214 1 233 1
252 1 271
      1 6   1 25  1 44  1 63  1 82  1 101 1 120 1 139 1 158 1 177 1 196 1 215 1 234 1
253 1 272
      1 7   1 26  1 45  1 64  1 83  1 102 1 121 1 140 1 159 1 178 1 197 1 216 1 235 1
254 1 273
      1 8   1 27  1 46  1 65  1 84  1 103 1 122 1 141 1 160 1 179 1 198 1 217 1 236 1
255 1 274
      1 9   1 28  1 47  1 66  1 85  1 104 1 123 1 142 1 161 1 180 1 199 1 218 1 237 1
256 1 275
      1 10  1 29  1 48  1 67  1 86  1 105 1 124 1 143 1 162 1 181 1 200 1 219 1 238 1
257 1 276
      1 11  1 30  1 49  1 68  1 87  1 106 1 125 1 144 1 163 1 182 1 201 1 220 1 239 1
258
      1 12  1 31  1 50  1 69  1 88  1 107 1 126 1 145 1 164 1 183 1 202 1 221 1 240 1
259
      1 13  1 32  1 51  1 70  1 89  1 108 1 127 1 146 1 165 1 184 1 203 1 222 1 241 1
260
      1 14  1 33  1 52  1 71  1 90  1 109 1 128 1 147 1 166 1 185 1 204 1 223 1 242 1
261
      1 15  1 34  1 53  1 72  1 91  1 110 1 129 1 148 1 167 1 186 1 205 1 224 1 243 1
262
      1 16  1 35  1 54  1 73  1 92  1 111 1 130 1 149 1 168 1 187 1 206 1 225 1 244 1
263
```

```
      1 17  1 36  1 55  1 74  1 93  1 112 1 131 1 150 1 169 1 188 1 207 1 226 1 245 1
264
      1 18  1 37  1 56  1 75  1 94  1 113 1 132 1 151 1 170 1 189 1 208 1 227 1 246 1
265
      1 19  1 38  1 57  1 76  1 95  1 114 1 133 1 152 1 171 1 190 1 209 1 228 1 247 1
266
```


G
-2G
TANGENT
-4G
8G
REFLECT ON X-AXIS
REFLECT ON X-AXIS
REFLECT ON X-AXIS
4G
TANGENT
-8G
TANGENT
2G

Schnorr Signature

- Signing
  - Let = g^k $r = g^k$
  - Let e = H(r || M) = H(|| M) $r = g^k$
  - Let = $s = k - xe$ $k - xe$ xe = H(r || M) -
  - The signature is (, s = $k - xe$ e = H(r || M))
- Verify
  - Let $r = g^k{}_v = g^s y^{ee}$
  - Let e = H(r || M)$_v$ = H($r = g^k v$ || M)
  - If e = H(r || M)$_v$ = e = H(r || M)

    the signature is valid.

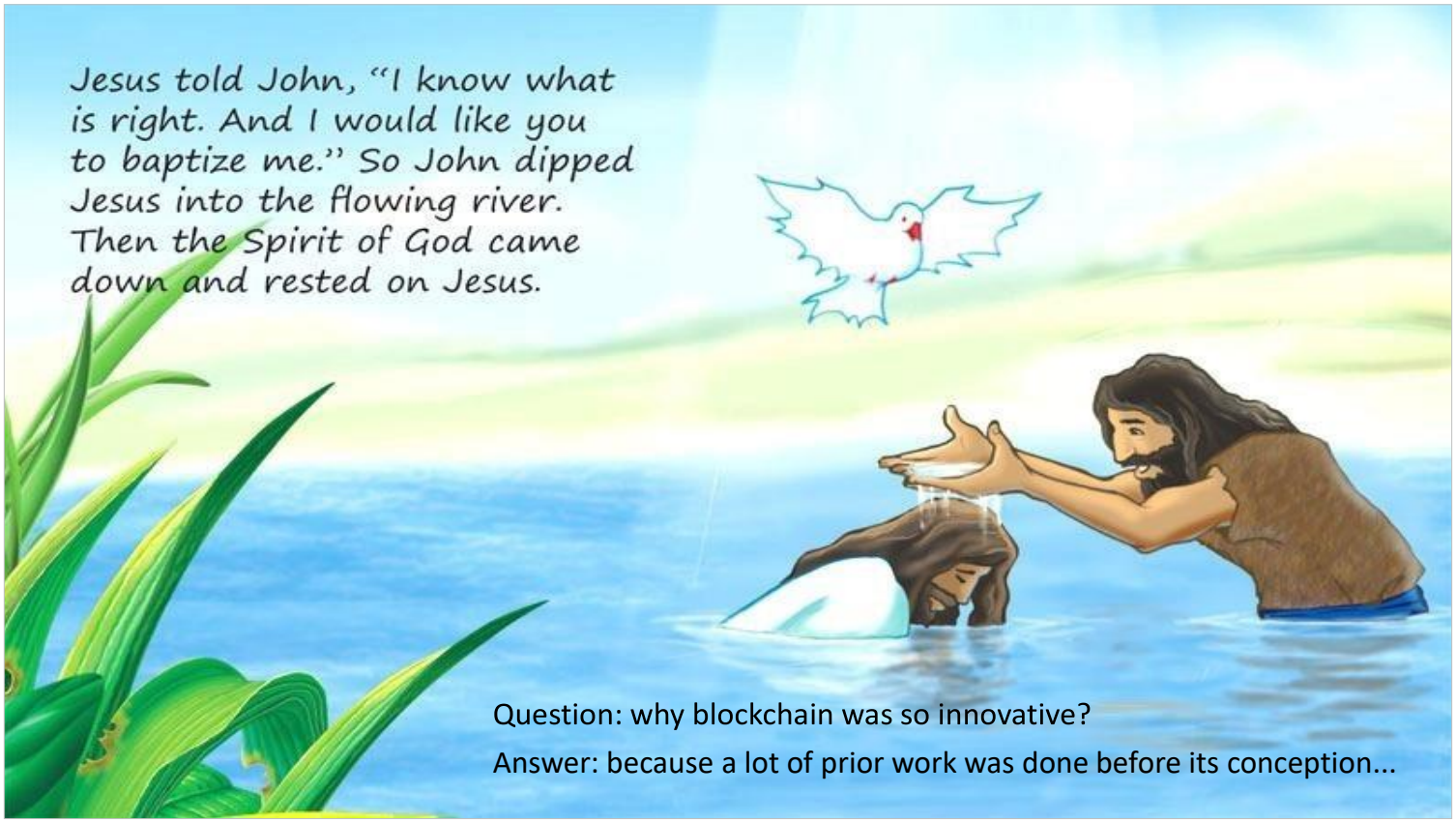
Talk 1: Elliptic curve and what was achieved before blockchain

Ask yourself, which of the following are unique to blockchain?

a) integrity (data can be tested against changes).

b) authenticity (only the keyholder could have done it).

c) Non-repudiation (undeniable proof);



Question: why blockchain was so innovative?

Answer: because a lot of prior work was done before its conception...

Properties of a digital signature scheme

a) integrity (data can be tested against changes).

b) authenticity (only the keyholder could've done it).

c) Non-repudiation (the keyholder couldn't deny having done it);

Please wait a minute our support staff will be online.Please wait a minute our support staff will be online. You are chatting with Alice.You are chatting with Alice.

Alice: Hi how can I help you today.

Bob: I wish to change my number to a new SIM card. The card number is Bob: I wish to change my number to a new SIM card. The card number is 8972837283748273381.8972837283748273381.

Alice: Great I can certainly help you with that. Can you tell me your name, Alice: Great I can certainly help you with that. Can you tell me your name, date of birth and your support code?date of birth and your support code?

Bob: Sure. 08-AUG-1988, Bob Boon, my secret code is Satoshi Nakamoto.Bob: Sure. 08-AUG-1988, Bob Boon, my secret code is Satoshi Nakamoto.

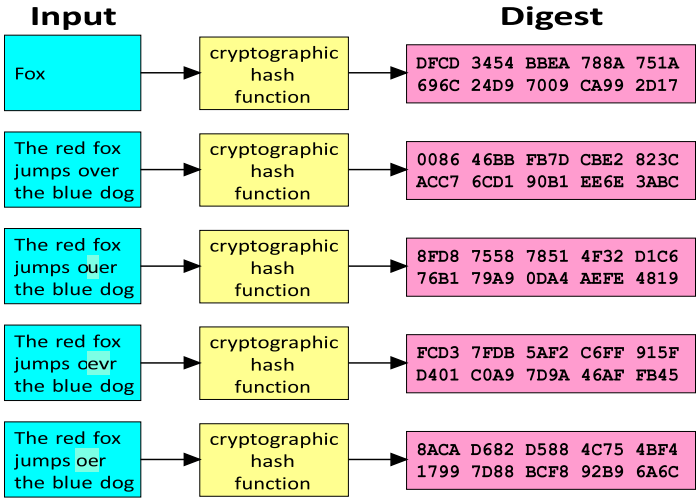
Alice: Perfect. Thank you Mr Boon. Your number is transferred to the new Alice: Perfect. Thank you Mr Boon. Your number is transferred to the new SIM card. What else can I do for you today?SIM card. What else can I do for you today?

Bob: That's all good thanks!Bob: That's all good thanks!

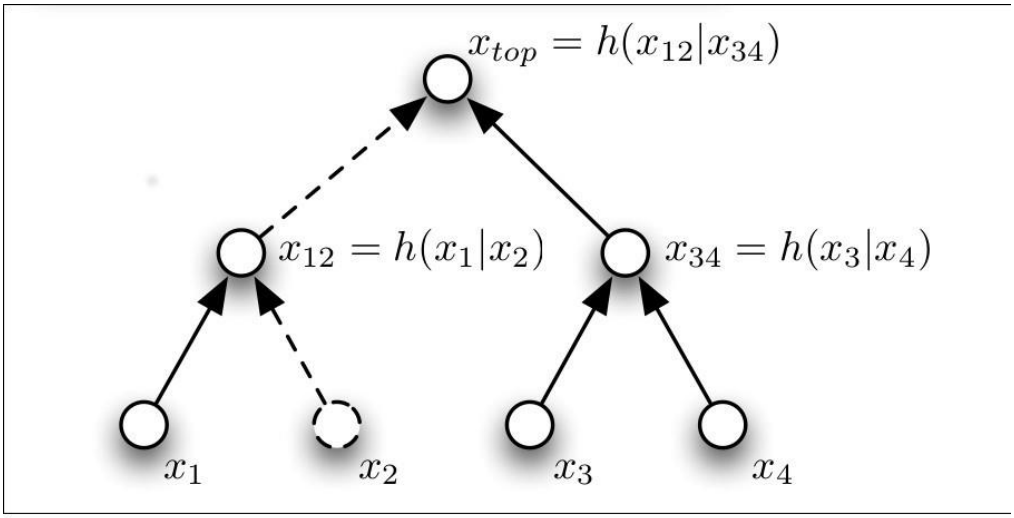
Alice: Have a nice day! Please rate my service from 1 to 5.Alice: Have a nice day! Please rate my service from 1 to 5.

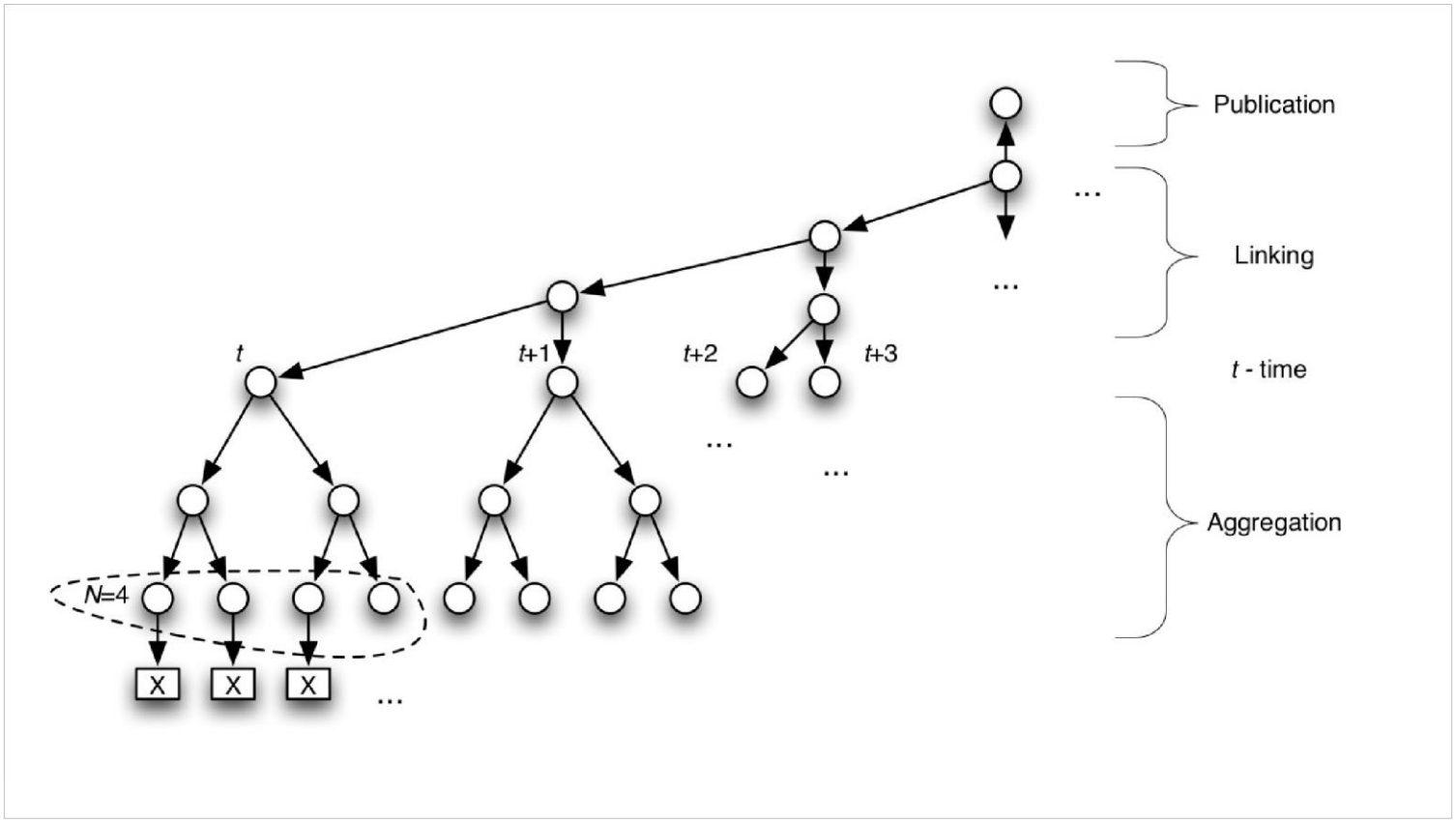
[Bob left the conversation[Bob left the conversation]]

Data integrity 1: Hash Function



https://en.wikipedia.org/wiki/File:Cryptographic_Hash_Function.svg


$x_{top} = h(x_{12}|x_{34})$
$x_{12} = h(x_1|x_2)$
$x_{34} = h(x_3|x_4)$
x_1
x_2
x_3
x_4

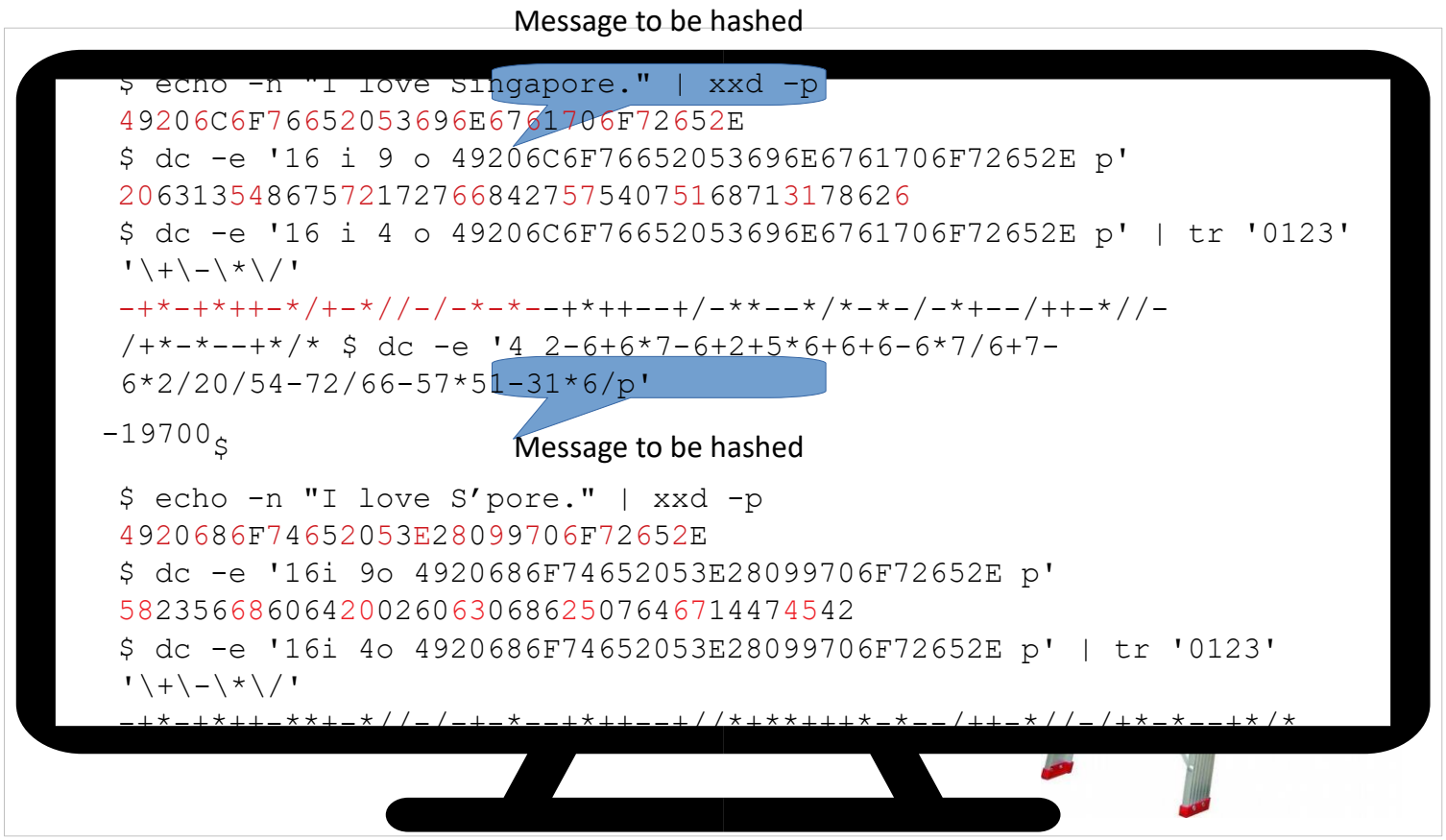

Publication
Linking
t - time
Aggregation
t
t+1
t+2
t+3
N=4
X
X
X

A ladder approach to one way function and hash function

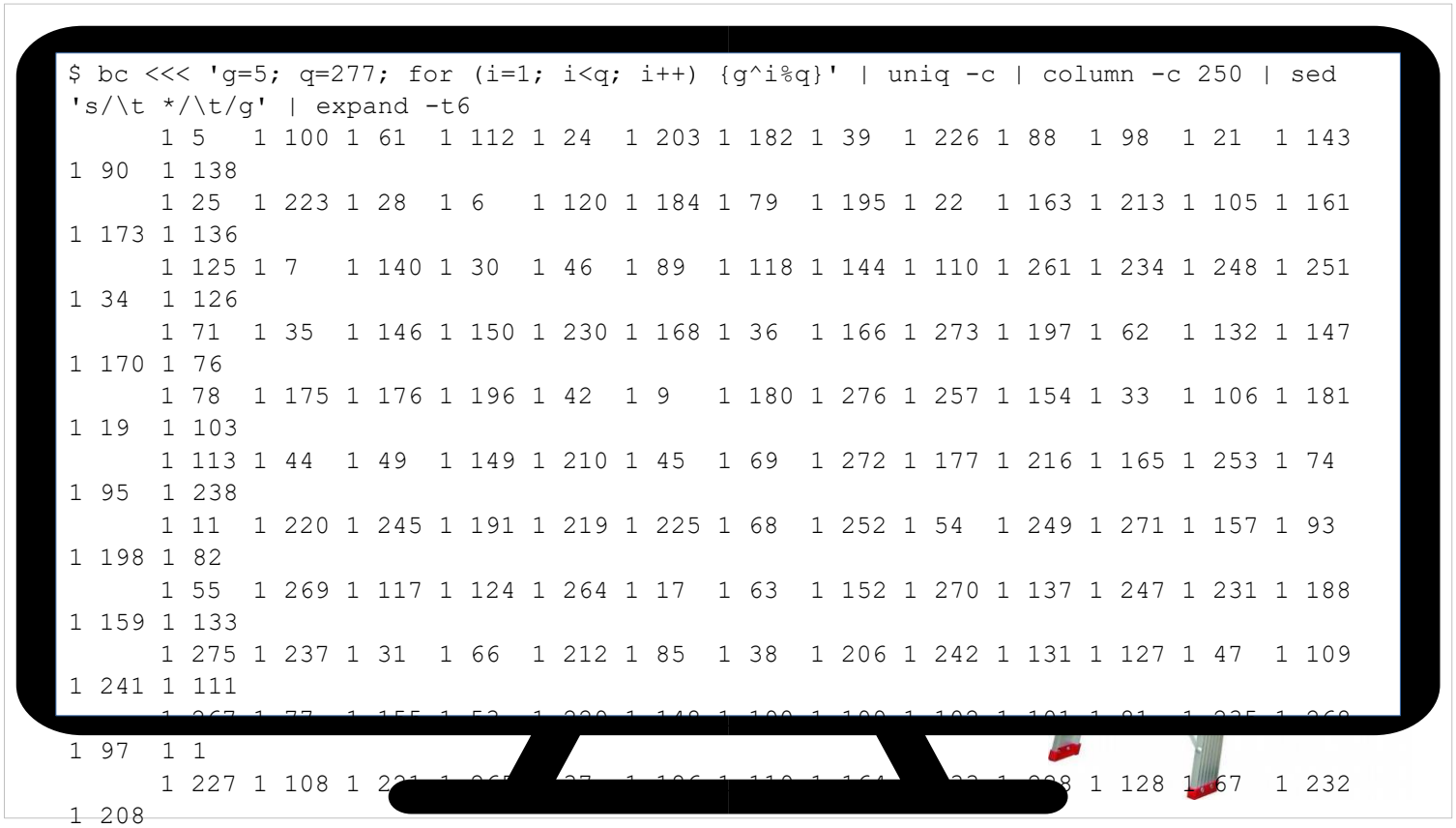
Jacob's Dream, 1705, James Thornhill

$$387420489 = x^x$$

We're here

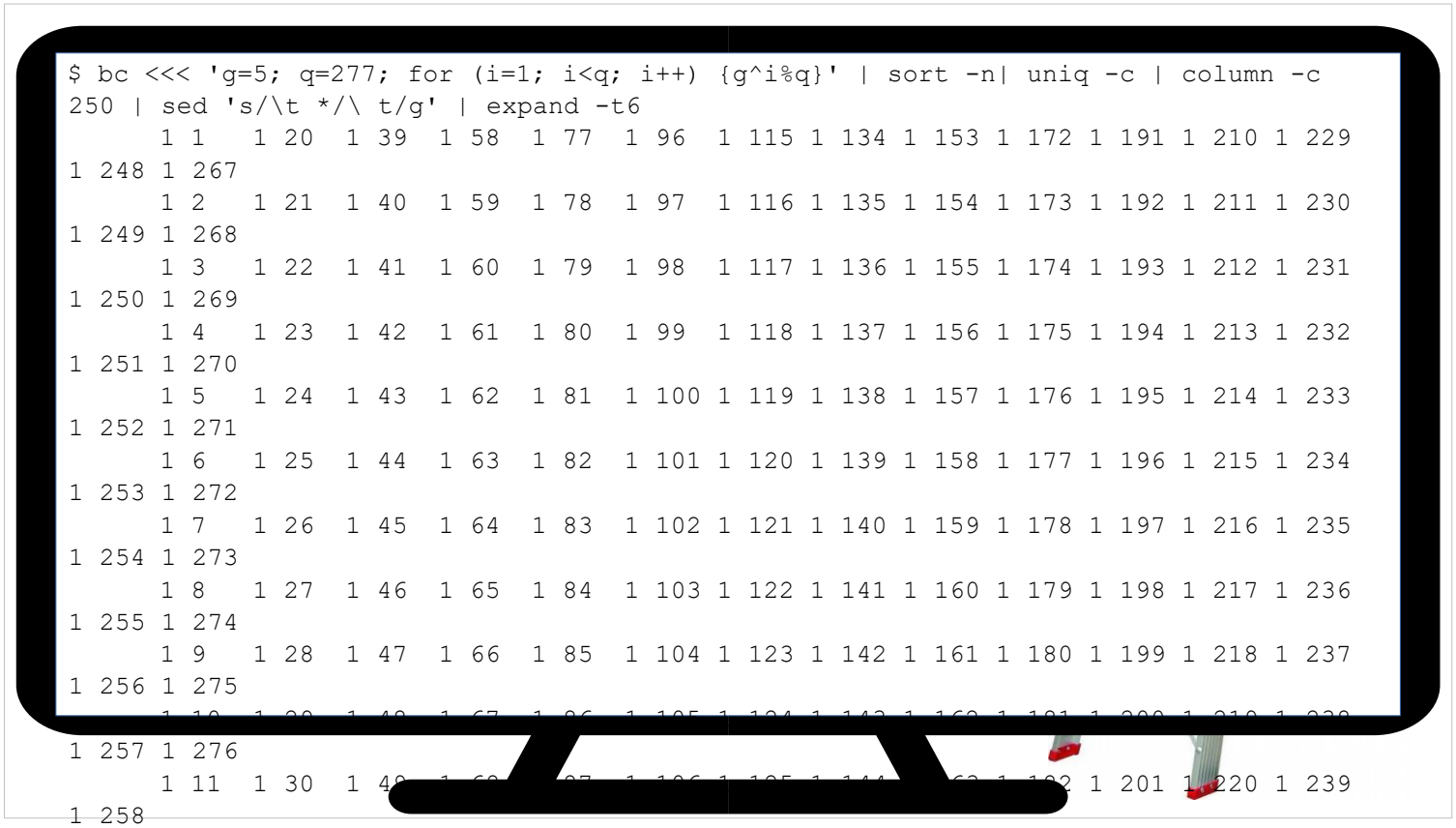

Message to be hashed
$ echo -n "I love Singapore." | xxd -p
49206C6F76652053696E676170 6F72652E
$ dc -e '16 i 9 o 49206C6F76652053696E6761706F72652E p'
20631354867572172766842757540751687131786 26
$ dc -e '16 i 4 o 49206C6F76652053696E6761706F72652E p' | tr '0123'
'\+\-*\/'
-+*-+*++-*/+-*//-/-*-*--+*++--+/-**--*/*-*-/-*+--/++-*//-
/+*-*--+*/* $ dc -e '4 2-6+6*7-6+2+5*6+6+6-6*7/6+7-
6*2/20/54-72/66-57*51-31*6/p'
-19700$
Message to be hashed
$ echo -n "I love S'pore." | xxd -p
4920686F74652053E28099706F72652E
$ dc -e '16i 9o 4920686F74652053E28099706F72652E p'
58235668606420026063068625076467144745 42
$ dc -e '16i 4o 4920686F74652053E28099706F72652E p' | tr '0123'
'\+\-*\/'

```
$ dc -e '4 2-6+6*7-6+2*5+E+8-9*7*6-7*6/2/58-68/20-63+25-
67*45-p' 4846
```



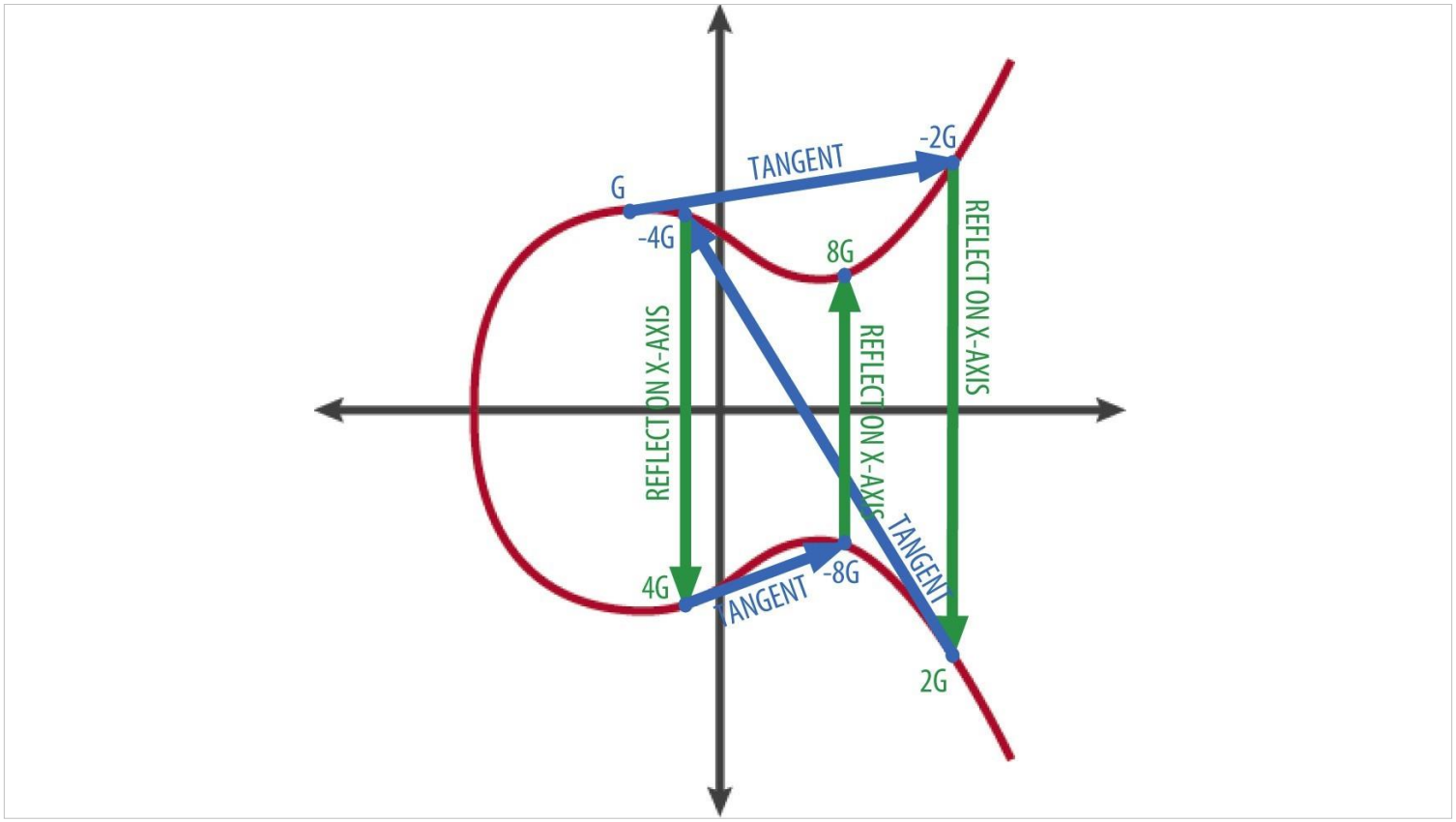
```
$ bc <<< 'g=5; q=277; for (i=1; i<q; i++) {g^i%q}' | uniq -c | column -c 250 | sed
's/\t */\t/g' | expand -t6
      1 5   1 100 1 61  1 112 1 24  1 203 1 182 1 39  1 226 1 88  1 98  1 21  1 143
1 90  1 138
      1 25  1 223 1 28  1 6   1 120 1 184 1 79  1 195 1 22  1 163 1 213 1 105 1 161
1 173 1 136
      1 125 1 7   1 140 1 30  1 46  1 89  1 118 1 144 1 110 1 261 1 234 1 248 1 251
1 34  1 126
      1 71  1 35  1 146 1 150 1 230 1 168 1 36  1 166 1 273 1 197 1 62  1 132 1 147
1 170 1 76
      1 78  1 175 1 176 1 196 1 42  1 9   1 180 1 276 1 257 1 154 1 33  1 106 1 181
1 19  1 103
      1 113 1 44  1 49  1 149 1 210 1 45  1 69  1 272 1 177 1 216 1 165 1 253 1 74
1 95  1 238
      1 11  1 220 1 245 1 191 1 219 1 225 1 68  1 252 1 54  1 249 1 271 1 157 1 93
1 198 1 82
      1 55  1 269 1 117 1 124 1 264 1 17  1 63  1 152 1 270 1 137 1 247 1 231 1 188
1 159 1 133
      1 275 1 237 1 31  1 66  1 212 1 85  1 38  1 206 1 242 1 131 1 127 1 47  1 109
1 241 1 111
      1 267 1 77  1 155 1 52  1 200 1 142 1 190 1 199 1 102 1 101 1 91  1 235 1 260
1 97  1 1
      1 227 1 108 1 221 1 265 1 37  1 196 1 110 1 164 1 33  1 228 1 128 1 67  1 232
1 208
```

```
      1 27  1 263 1 274 1 217 1 185 1 99  1 41  1 266 1 57  1 32  1 86  1 58  1 52
1 209
      1 135 1 207 1 262 1 254 1 94  1 218 1 205 1 222 1 8   1 160 1 153 1 13  1 260
1 214
      1 121 1 204 1 202 1 162 1 193 1 259 1 194 1 2   1 40  1 246 1 211 1 65  1 192
1 239
      1 51  1 189 1 179 1 256 1 134 1 187 1 139 1 10  1 200 1 122 1 224 1 48  1 129
1 87
      1 255 1 114 1 64  1 172 1 116 1 104 1 141 1 50  1 169 1 56  1 12  1 240 1 91
1 158
      1 167 1 16  1 43  1 29  1 26  1 243 1 151 1 250 1 14  1 3   1 60  1 92  1 178
1 236
      1 4   1 80  1 215 1 145 1 130 1 107 1 201 1 142 1 70  1 15  1 23  1 183 1 59
1 72
      1 20  1 123 1 244 1 171 1 96  1 258 1 174 1 156 1 73  1 75  1 115 1 84  1 18
1 83
```



```
$ bc <<< 'g=5; q=277; for (i=1; i<q; i++) {g^i%q}' | sort -n| uniq -c | column -c
250 | sed 's/\t */\ t/g' | expand -t6
      1 1   1 20  1 39  1 58  1 77  1 96  1 115 1 134 1 153 1 172 1 191 1 210 1 229
1 248 1 267
      1 2   1 21  1 40  1 59  1 78  1 97  1 116 1 135 1 154 1 173 1 192 1 211 1 230
1 249 1 268
      1 3   1 22  1 41  1 60  1 79  1 98  1 117 1 136 1 155 1 174 1 193 1 212 1 231
1 250 1 269
      1 4   1 23  1 42  1 61  1 80  1 99  1 118 1 137 1 156 1 175 1 194 1 213 1 232
1 251 1 270
      1 5   1 24  1 43  1 62  1 81  1 100 1 119 1 138 1 157 1 176 1 195 1 214 1 233
1 252 1 271
      1 6   1 25  1 44  1 63  1 82  1 101 1 120 1 139 1 158 1 177 1 196 1 215 1 234
1 253 1 272
      1 7   1 26  1 45  1 64  1 83  1 102 1 121 1 140 1 159 1 178 1 197 1 216 1 235
1 254 1 273
      1 8   1 27  1 46  1 65  1 84  1 103 1 122 1 141 1 160 1 179 1 198 1 217 1 236
1 255 1 274
      1 9   1 28  1 47  1 66  1 85  1 104 1 123 1 142 1 161 1 180 1 199 1 218 1 237
1 256 1 275
      1 10  1 29  1 48  1 67  1 86  1 105 1 124 1 143 1 162 1 181 1 200 1 219 1 238
1 257 1 276
      1 11  1 30  1 49  1 68  1 87  1 106 1 125 1 144 1 163 1 182 1 201 1 220 1 239
1 258
```

```
      1 12  1 31  1 50  1 69  1 88  1 107 1 126 1 145 1 164 1 183 1 202 1 221 1 240
1 259
      1 13  1 32  1 51  1 70  1 89  1 108 1 127 1 146 1 165 1 184 1 203 1 222 1 241
1 260
      1 14  1 33  1 52  1 71  1 90  1 109 1 128 1 147 1 166 1 185 1 204 1 223 1 242
1 261
      1 15  1 34  1 53  1 72  1 91  1 110 1 129 1 148 1 167 1 186 1 205 1 224 1 243
1 262
      1 16  1 35  1 54  1 73  1 92  1 111 1 130 1 149 1 168 1 187 1 206 1 225 1 244
1 263
      1 17  1 36  1 55  1 74  1 93  1 112 1 131 1 150 1 169 1 188 1 207 1 226 1 245
1 264
      1 18  1 37  1 56  1 75  1 94  1 113 1 132 1 151 1 170 1 189 1 208 1 227 1 246
1 265
      1 19  1 38  1 57  1 76  1 95  1 114 1 133 1 152 1 171 1 190 1 209 1 228 1 247
1 266
```


G
TANGENT
-2G
-4G
8G
REFLECT ON X-AXIS
REFLECT ON X-AXIS
REFLECT ON X-AXIS
TANGENT
4G
TANGENT
-8G
2G

Schnorr Signature

- Signing
  - Let = g^k r = g^k
  - Let e = H(r || M) = H(|| M) r = g^k
  - Let = s = k - xe k - xe xe = H(r || M) -
- Verify
  - Let r = $g^k{}_v$ = $g^s y^{ee}$
  - Let e = H(r || M)$_v$ = H(r = $g^k v$ || M)
  - If e = H(r || M)$_v$ = e = H(r || M)

- The signature is (, s = k - xe e = H(r || M)) the signature is valid.