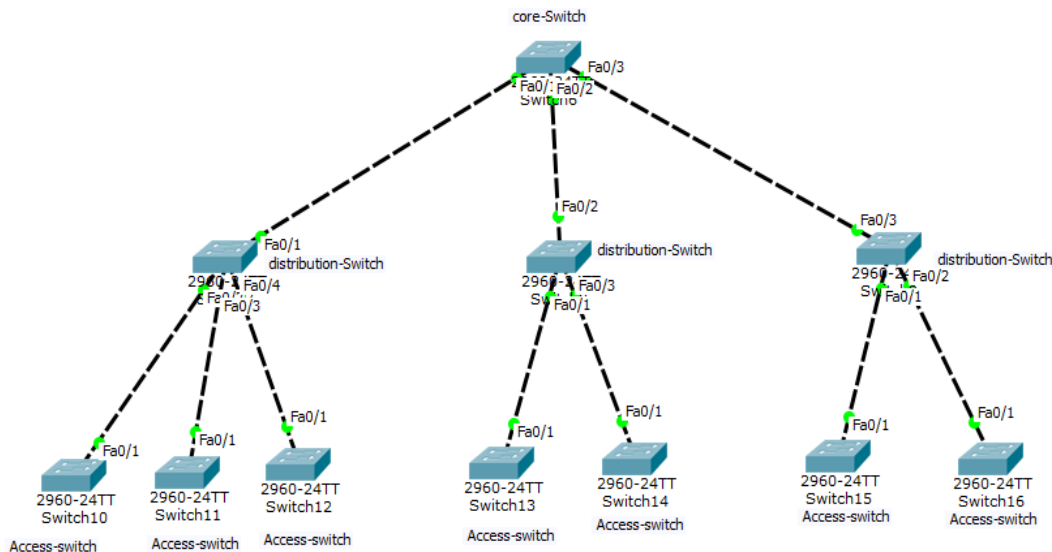


## Spanning Tree Protocol

(STP)

(IEEE 802.1d)

❖ نبدأ بطريقة التوسع بالشبكات المعروفة باسم ( Hub&&Spoke )، والموضحة بالشكل التالي :

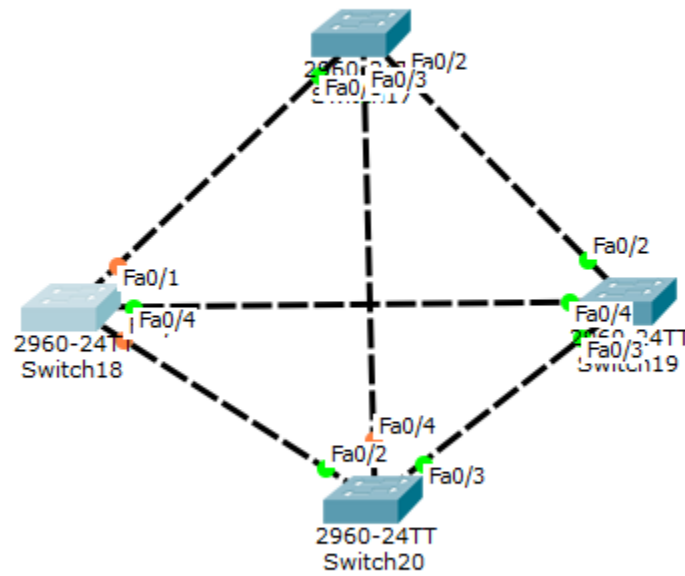


المشكلة في هذا التصميم أنه إذا قطعت وصلة من الوصلات أو خرج منفذ عن العمل (أصبح بحالة Down ) ، فإنه سيسبب خروج قسم كامل من الشبكة خارج الخدمة .

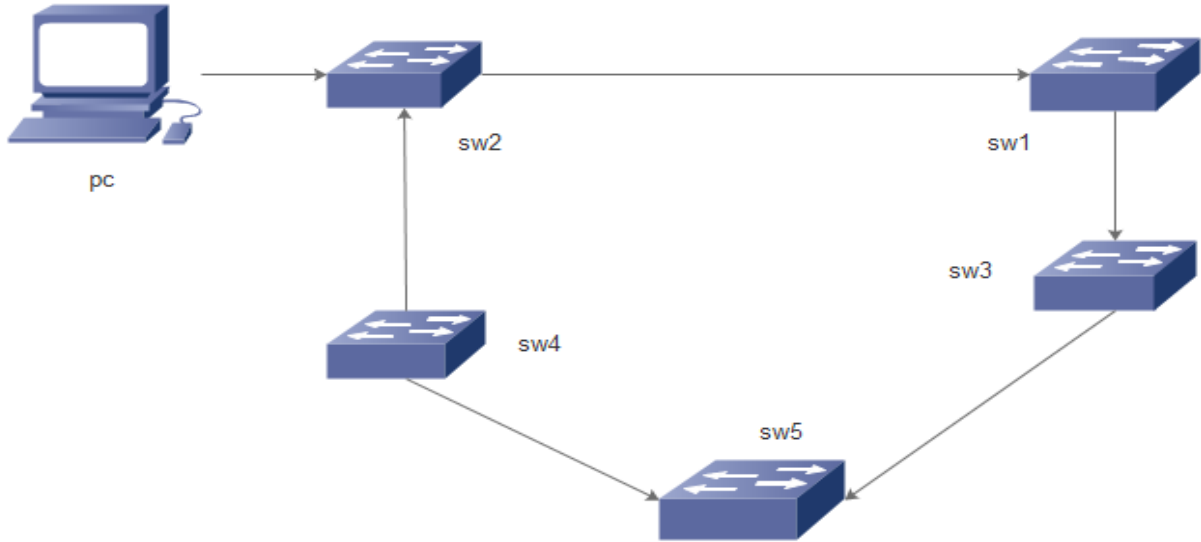
فإذا كانت الشركات تقدم خدمات مهمة (online) فإن طريقة التصميم Hub & Spoke ، لا تفي بالغرض ولا يمكن استخدام هذا التصميم، لأن خروج قسم كامل من الشركة عن العمل يكون بمثابة كارثة للشركة .

❖ فكان الحل لهذه المشكلة مع نمط التوسع في الشبكات والذي هو ( Full mesh ) :

يستند هذا النمط على أن أي تجهيز يجب أن يتصل مع أي تجهيز آخر في الشبكة دون المرور بجهاز وسيط ،وبالتالي إذا قطعت وصلة من الوصلات لم تتأثر الشبكة، ونتج عن هذا التصميم موثوقية عالية في الشبكة، كما هو موضح في الشكل التالي :



❖ لكن طريقة التصميم Full Mesh تسبب الى تشكل الحلقات المغلقة وهذا ما يعرف بال ( Looping ) ، لتوضيح هذه المشكلة لنأخذ المثال التالي، وهي عبارة عن إحدى الحلقات المغلقة المؤلفة من 5 switch ، كما في الشكل التالي :



لنفرض أن ال PC قام بإرسال باكيت Broadcast ال Sw2

يقوم ال Sw2 بعملية غمر (flooding) للباكيت الى كل المنافذ باستثناء المنفذ الوارد منه

ستتجه نسخة من ال Broadcast packet ال Sw1 ونسخة أخرى الى Sw 4

النسخة الأولى بعد أن تصل ال Sw 1 سيقوم بغمرها وبالتالي تصل الى Sw 5 وهكذا الى أن تعود الى Sw2 ليقوم بإعادة غمرها من جديد ، لتتكرر الحلقة ( loop )

والنسخة الثانية من البايت التي تصل الى Sw4 سيقوم بغمرها وبالتالي تصل الى Sw5 ثم الى Sw 3 وبالتالي تعود أيضا الى Sw2 ليقوم ببثها أيضا جديد لتتكرر الحلقة (loop) وهكذا ستتكرر الحلقات الى ما لانهاية ، الى أن تؤدي الى توقف الشبكة بالكامل عن العمل ، وتعرف هذه المشكلة بعاصفة البث العام (Broadcast Storm).

❖ فكان الحل الأمثل للإنتهاء من مشكلة ال (Broadcasting Storm) مع المحافظة على التصميم Full Mesh ، هو البرتوكول STP .

❖ قبل البدء بخوارزمية ال STP ، سنتعرف على مفهوم ال BPDUs وهو اختصارا ل ( Bridge Packet Data Unit ) ، وهي عبارة عن إشارو يقوم كل switch بإرسالها عند تشغيله ، تحتوي على معلوماتين أساسيتين عن هذا ال Switch :

1. Priority Number الخاص بال Switch (رقم الأولوية)

2. MAC Address الخاص بال Switch

إن مجموع هذين الرقمين في أي Switch يدعى ال BID (Bridge Identity) (هوية ال Switch)

❖ تعتمد خوارزمية البرتوكول STP على ثلاث عمليات انتخاب :

1. **Root Bridge Election** عملية انتخاب السويتش الجذر .

2. **Root Port Election** عملية انتخاب المنفذ الجذر .

3. **Designated Port Election** .

🌈 **Root Bridge Election** عملية انتخاب السويتش الجذر :

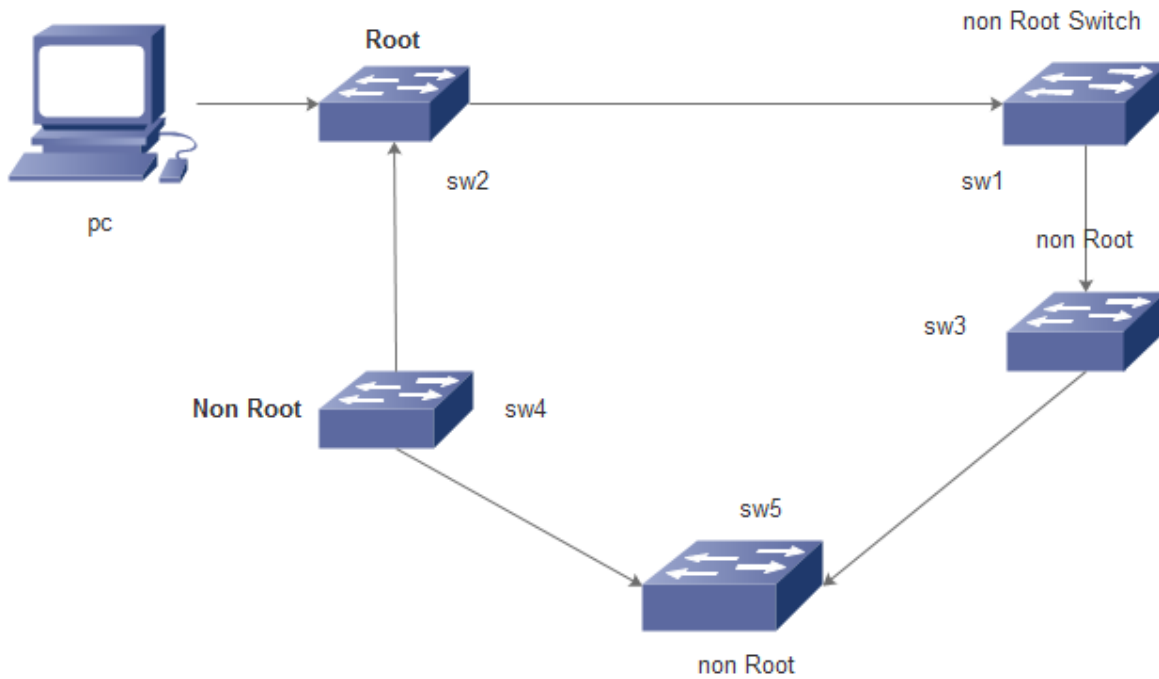
إن ال Switch الذي يملك الرقم BID الأصغر في الحلقة هو الذي سوف يصبح ال Root .

ملاحظة : عندما تقوم Cisco بتصنيع ال Switch فإن كافة السويتشات تمتلك نفس ال Priority Number وهو  $PN=32768$  ، بالتالي بشكل افتراضي فإن السويتش الذي يملك ال MAC الأقل عمليا هو الذي سوف يصبح Root .

✓ كيف تتم عملية انتخاب ال Root Bridge :

عندما أقوم بتشغيل السويتشات كل Sw يعتبر نفسه هو ال Root ويرسل ال BID الخاص به، كلما أتت ل Switch رقم BID أصغر منه من رقم ال BID الخاص به، يعلم أنه ليس Root ويتوقف عن إرسال ال BPDUs، وهكذا يستمر سويتش واحد الذي يملك أقل BID بإرسال BPDUs في الحلقة المغلقة فقط ، ويكون هو ال Root .

لنفترض في مثالنا أن ال SW2 هو ال Root ، فإن باقي السويتشات يتم تسميتها **Non Root Switch** ، كما يلي :



## Root Port Election انتخاب البورت الجذر :

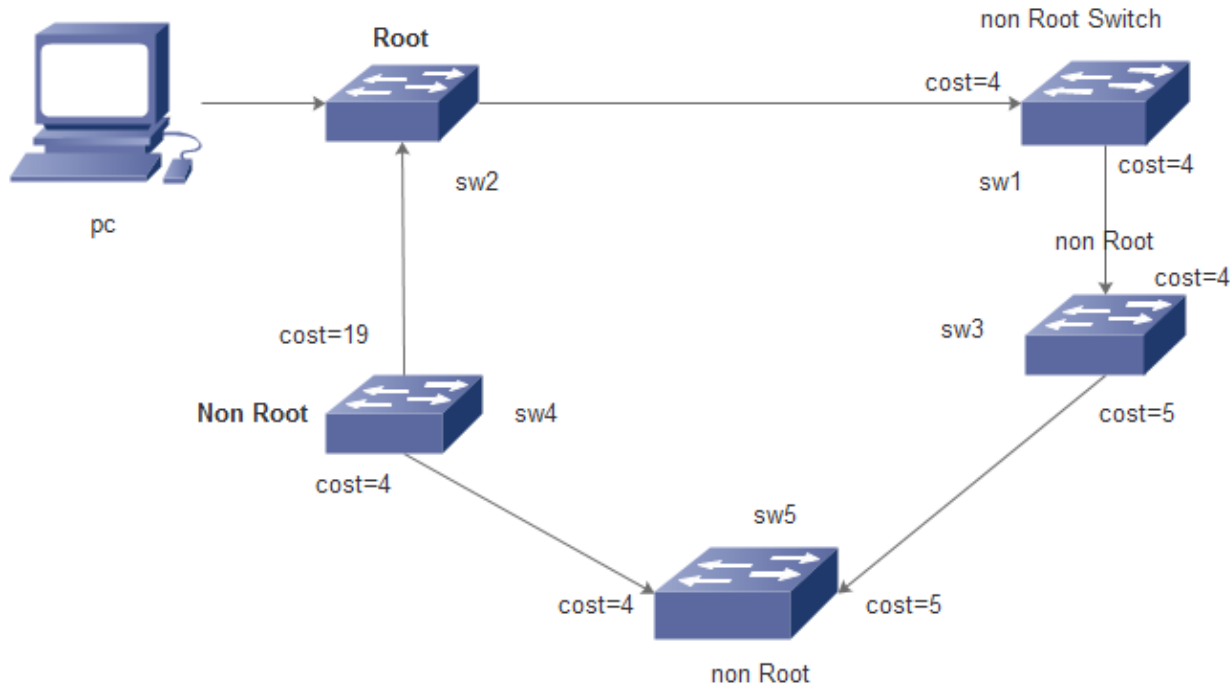
هو منفذ موجود على Non Root bridge والذي يسمح بالوصول الى ال Root بأقل تكلفة ممكنة .

ملاحظة : كل منفذ له تكلفة تتعلق بسرعة ذلك المنفذ ، وبالتالي دخول إشارة ال BPDU بمنفذ سرعته 10mbps تختلف عن دخولها بمنفذ سرعته 100 mbps ، وبناءا على ذلك يمكننا استنتاج الجدول التالي والذي يسمى ( STP Cost Table ) :

Data rate	STP Cost
4 Mbps	250
10 Mbps	100
16 Mbps	62
100 Mbps	19
1 Gbps	4
2 Gbps	3
10 Gbps	2

ملاحظة : ال Cost يحسب عند دخول BPDU بمنفذ وليس عند خروجها من المنفذ .

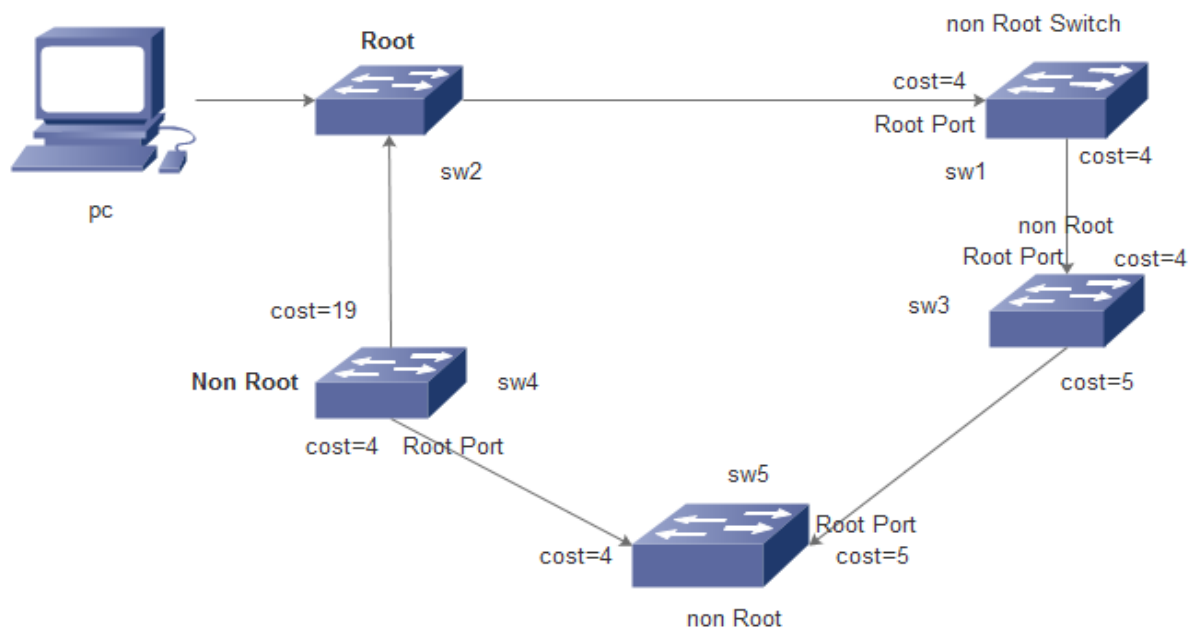
لنفرض أن لدينا ال Cost للمنفذ موضح بالشكل التالي :



لنقوم بانتخاب المنفذ الجذر (Root Port) على كل non Root Switch :

نبدأ ب Sw1 ستأتيه إشارة BPDUs قامت بحساب الكلفة (4) من المنفذ الأول و BPDUs قامت بحساب (32=4+5+4+19) من جهة المنفذ الثاني وبالتالي ال Root Port على هذا السويتش هو المنفذ الأول لأنه المنفذ الذي يوصل الى ال Root bridge بأقل تكلفة، نقوم بنفس العملية على السويتشات التي هي (non Root Switch)، فينتج لدينا ال Root Port كما في الشكل التالي :





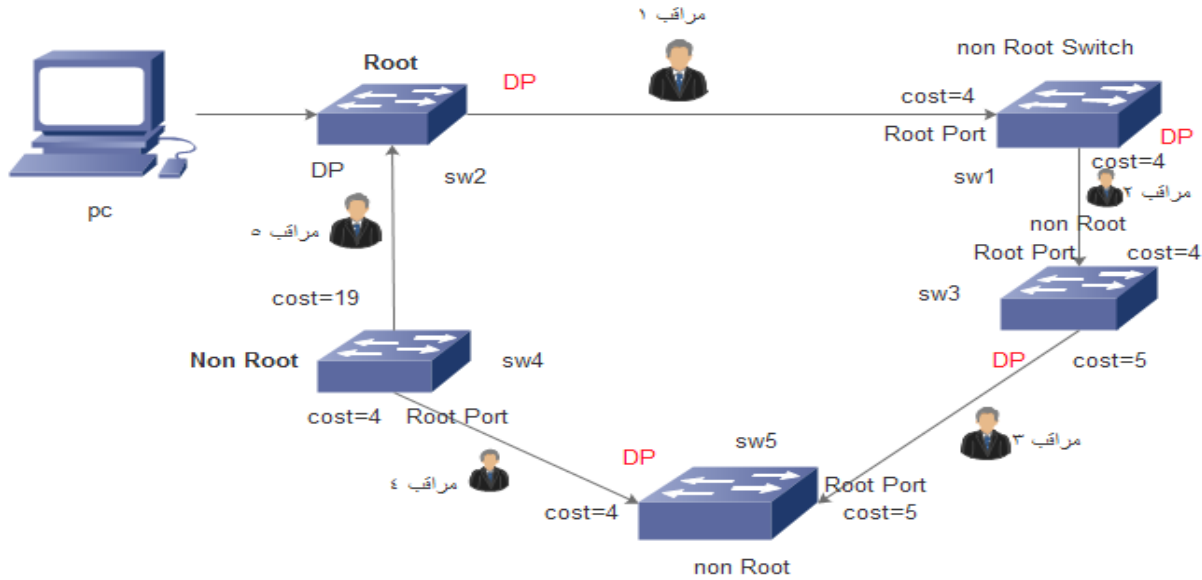
### : Designated port Election

يتم انتخاب هذا البورت من وجهة نظر المقاطع الشبكية (الوصلات) بين السويتشات باعتبارها كائنات تنقل ال Data وتؤثر عليها بالنسبة الى مراقب يقف في منتصف الوصلة، بحيث تسمح له بالوصول الى ال Root Switch بأقل تكلفة ممكنة، ويمكن أن يوجد على Root Switch و non Root Switch ، مما سبق نستنتج أن كل منافذ ال Root هي Designated port .

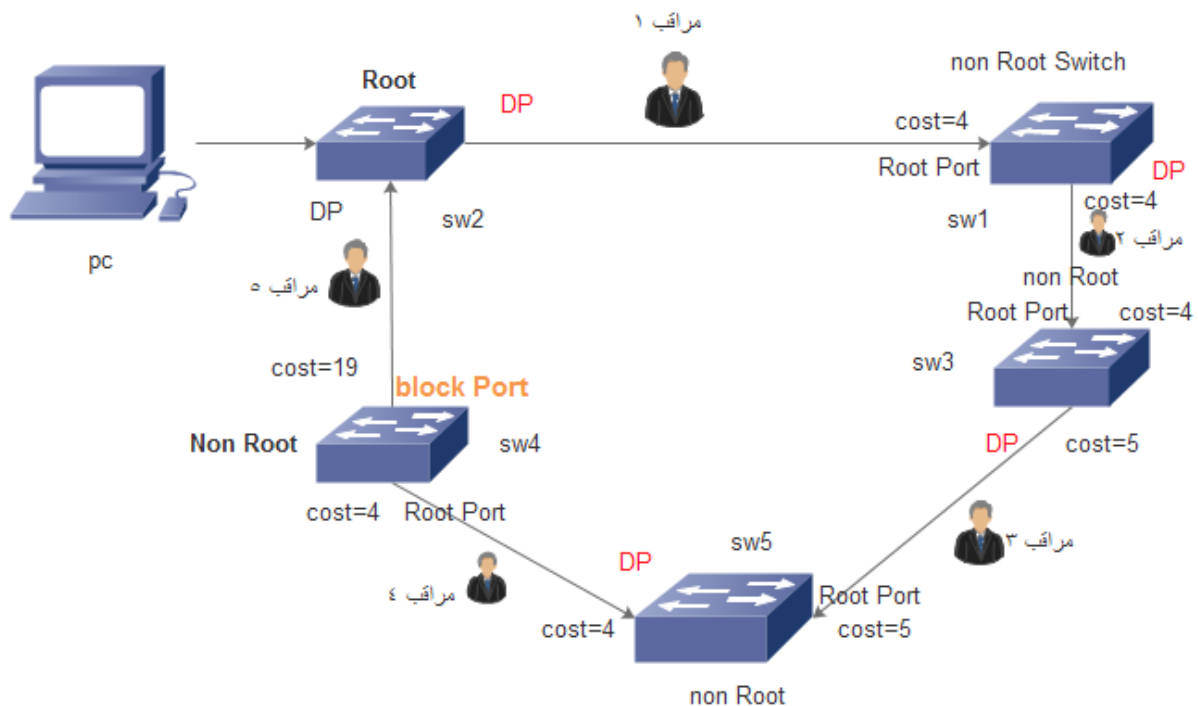
✓ لنقوم بانتخاب ال Designated port في مثالنا:

بالنسبة للمراقب 1: تأتيه رسالة PBDU من Sw2 تحتوي على تكلفة ( 0 )، ومن Sw1 إشارة PBDU تحتوي على تكلفة (  $19+4+5+4=32$  )، وبالتالي DP هو المنفذ من جهة ال SW2 وهو ال Root .

وهكذا يتم انتخاب جميع المنافذ ال DP ، كما هو موضح على الشكل :



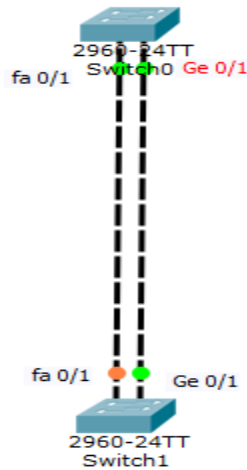
- نلاحظ أنه في كل حلقة ينتج منفذ واحد ليس Root Port ولا Designated port، تقوم خوارزمية ال STP بتحويله الى حالة (Block Port)، وهكذا حالة المنفذ blocking قامت بكسر الحلقة looping، وبالتالي تخلصنا من مشكلة عاصفة البث العام، كما هو موضح بالشكل :



- إن المنفذ في حالة **blocking** بالنسبة لل **STP** ، يصبح يسمح لنوع واحد من الإشارات بالمرور وهي إشارات ال **BPDUs**، ويوقف أي عمليات بث أخرى سواء **Data** ، أو غيرها .
- في حالة أي كسر للحلقة (حدوث مشكلة في أحد المنافذ أو الوصلات في الحلقة )، ينتقل المنفذ الذي في حالة **blocking** الى حالة **Forward** ، وهكذا لم نخسر فائدة التصميم **Full-Mesh**.
- لماذا نحن بحاجة لجعل المنفذ يبقى يمرر رسائل ال **BPDUs**، لأنه عندما سيمضي زمن معين ولا يمرر ال **BPDUs** ، هذا يعني أنه يوجد احتمالين :
  1. إذا انقطعت حلقة كيف سيعلم المنفذ الذي في حالة **Blocking** أنه تم قطع الحلقة، عندما سيمضي زمن معين ولا يمرر ال **BPDUs**، سيكون الحل بتحويل المنفذ الذي في حالة **Blocking** الى **Forward** .
  2. إما أن ال **Root** قد توقف عن العمل ، لنفرض أن ال **Root** أصبح **Down** :
    - ينتظر هذا المنفذ **2sec**
    - بعد **2sec** تعلم بقية السويتشات أن ال **Root Switch** أصبح **Down** .
    - تمنح خوارزمية ال **STP** (**20sec**) لأنها تضع احتمال أن ال **Root** قد قامت بعملية ال **Restart** ،
    - إذا مضت **2sec** وال **20sec** ولم تصل ال **BPDUs** وبالتالي تبدأ عملية الانتخاب .
    - حيث تمر بمرحلة مدتها **15sec**، تكون المنافذ فيها بحالة تسمى (**Listening**)، تتسمع فيها على ال **BID** الذي تقوم بإرساله بقية السويتشات، في نهاية هذا المرحلة يكون قد تم تحديد ال **Root Switch** .
    - ثم تبدأ مرحلة تكون فيها المنافذ في حالة (**learninig**)، في نهاية هذا المرحلة يكون قد تم تحديد (**Root Port**) ، وال (**Designated port**) .

نلاحظ أنه أصبح المجموع 52 ثانية ،لتنفيذ خوارزمية ال STP في حال حدوث مشكلة في ال Root Switch ، و30 ثانية بشكل افتراضي سيمر بها في كل عملية انتخاب .

❖ تطبيق عملي :لنفرض لدينا إحدى حلقات ال STP ، كما هو موضح :



الأمر **Show spanning-tree** :يعطي هذا الامر معلومات حول السويتش الذي أعمل عليه (Bridge ID) والسويتش ال **Root** ( **Root ID** ) في الحلقة ، ومعلومات حول المنافذ على السويتش وحالة كل منفذ ووالدور الذي يقوم به واولويته وتكلفة المنفذ ونوعه ، كما هو موضح :

```
Switch>enable
Switch#show spanning-tree
VLAN0001
  Spanning tree enabled protocol ieee
  Root ID    Priority    32769
             Address     0001.C95C.D202
             This bridge is the root
             Hello Time 2 sec  Max Age 20 sec  Forward Delay 15 sec

  Bridge ID  Priority    32769 (priority 32768 sys-id-ext 1)
             Address     0001.C95C.D202
             Hello Time 2 sec  Max Age 20 sec  Forward Delay 15 sec
             Aging Time  20
```

Interface	Role	Sts	Cost	Prio.Nbr	Type
Gi0/1	Desg	FWD	4	128.25	P2p
Fa0/1	Desg	FWD	19	128.1	P2p

❖ ملاحظة : في حال تشابه ال Root ID مع ال Bridge ID على سويتش معين، يكون هو ال

### . Root Switch

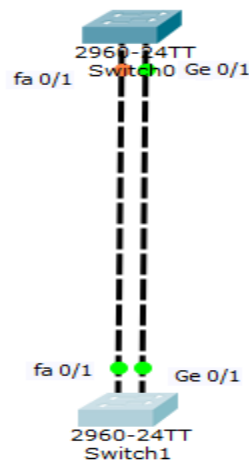
❖ ملاحظة : لنفرض أنه في التصميم السابق، لسبب ما ، نريد جعل ال Switch1 هو ال Root

Switch ، أقوم بتغيير ال Priority لل Switch1 الى ( 0 ) ، ندخل الى ال CLI ، وندخل

الأمر التالي :

```
Switch>enable
Switch#conf t
Enter configuration commands, one per line. End with CNTL/Z.
Switch(config)#spanning-tree vlan 1 priority 0
```

نلاحظ أن ال Switch1 أصبح ال Root كما في الشكل :



- ❖ **STP Fast Port** : المنفذ عند تعيينه في حالة **Fast Port** ، فإنه لن ينفذ خوارزمية ال **STP** ،  
وعند وصل الحاسب الى المنفذ (**Fast Port**) ، في نفس اللحظة يتحول الى حالة **Forward** ،  
مثلا نريد جعل المنفذ **Fa 0/1 (Fast Port)** ، كما يلي :

```
Switch>enable
Switch#conf t
Enter configuration commands, one per line. End with CNTL/Z.
Switch(config)#interface fastethernet 0/1
Switch(config-if)#spanning-tree portfast
%Warning: portfast should only be enabled on ports connected to a single
host. Connecting hubs, concentrators, switches, bridges, etc... to this
interface when portfast is enabled, can cause temporary bridging loops.
Use with CAUTION

%Portfast has been configured on FastEthernet0/1 but will only
have effect when the interface is in a non-trunking mode.
```